

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日

Date of Application: 2003年 2月19日

出願番号

Application Number: 特願2003-041838

[ST.10/C]:

[JP2003-041838]

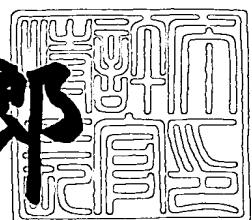
出願人

Applicant(s): 日本電気株式会社

2003年 6月20日

特許庁長官  
Commissioner,  
Japan Patent Office

太田 信一郎



出証番号 出証特2003-3048672

【書類名】 特許願  
【整理番号】 33509975  
【あて先】 特許庁長官殿  
【特記事項】 特許法第30条第1項の規定の適用を受けようとする特許出願  
【国際特許分類】 H04L 12/00  
H04M 7/00  
【発明者】  
【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内  
【氏名】 榎本 敦之  
【発明者】  
【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内  
【氏名】 鳥橋 正樹  
【発明者】  
【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内  
【氏名】 飛鷹 洋一  
【発明者】  
【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内  
【氏名】 岩田 淳  
【発明者】  
【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内  
【氏名】 渋谷 真  
【特許出願人】  
【識別番号】 000004237  
【氏名又は名称】 日本電気株式会社  
【代理人】  
【識別番号】 100093595  
【弁理士】  
【氏名又は名称】 松本 正夫

【先の出願に基づく優先権主張】

【出願番号】 特願2002-242621

【出願日】 平成14年 8月22日

【手数料の表示】

【予納台帳番号】 057794

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9303563

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 ネットワークシステム、スパニングツリー構成方法、スパニングツリー構成ノード、及びスパニングツリー構成プログラム

【特許請求の範囲】

【請求項1】 複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、

ネットワークの構成変更前のスパニングツリーを運用したまま、構成変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とするノード。

【請求項2】 前記ネットワークの構成変更が、ノードの追加又は削除やリンクトポジの変化であることを特徴とする請求項1に記載のノード。

【請求項3】 複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、

ネットワークのコスト変更時に、現スパニングツリーを運用したまま、リンクコスト変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とするノード。

【請求項4】 複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、

独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、

転送に利用しているスパニングツリーに対応したタグを返答するタグテーブルと、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、転送に利用するスパニングツリーを決定するツリーセレクタと、

宛先ごとのフレームの転送出力先を記録するフォワーディングテーブルと、

前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、

前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とするノード。

【請求項5】 前記ツリーセレクタが、  
転送を利用するスパニングツリーの切り替えを行うメインコントローラと、  
スパニングツリーの安定を示す指定した時間のタイマ満了を知らせる安定タイ  
マと、  
フレームに付加されたタグを削除するタグ削除器と、  
スパニングツリーの切り替えのための制御フレームを送信するGVRP送受信  
器と、  
フレームにタグを付加するタグ挿入器とを備えることを特徴とする請求項4に  
記載のノード。

【請求項6】 前記ツリーセレクタは、  
スパニングツリーの安定を示すフレームの到着間隔を判別するため、一定時間  
が経過するとタイマ満了通知を送信する到着間隔タイマを備えることを特徴とす  
る請求項5に記載のノード。

【請求項7】 前記ツリーセレクタは、  
リンクコストの算出を利用する指定時間のタイマ満了を知らせるコスト参照タ  
イマを備えることを特徴とする請求項4から請求項6の何れか1項に記載のノー  
ド。

【請求項8】 前記ツリーマネージャが、  
フレームに付加されたタグを削除するタグ削除器と、  
BPDUsを送受信するBPDUs送受信器と、  
フレームにタグを付加するタグ挿入器と、  
スパニングツリープロトコルに従ってスパニングツリーを作成するツリーコン  
トローラと、  
前記スパニングツリープロトコルで使用するパラメータを保持するツリーテー  
ブルとを備えることを特徴とする請求項4から請求項7の何れか1項に記載のノ  
ード。

【請求項9】 前記ツリーマネージャは、

通知されたリンクコストに所定の設定値を加算して返答するコスト操作器を備えることを特徴とする請求項8に記載のノード。

【請求項10】 リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタを備えることを特徴とする請求項4から請求項9の何れか1項に記載のノード。

【請求項11】 前記リンクコストを、利用状況に基づいて計算することを特徴とする請求項3に記載のノード。

【請求項12】 前記利用状況を、空き帯域と定義することを特徴とする請求項11に記載のノード。

【請求項13】 前記利用状況を、CPU負荷と定義することを特徴とする請求項11に記載のノード。

【請求項14】 複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、

ネットワーク内の各ノードがルートノードとなるスパニングツリーを生成し、宛先がルートノードとなるスパニングツリーを用いてフレームを転送することを特徴とするノード。

【請求項15】 複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、

独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、

転送に利用しているスパニングツリーに対応したタグを返答するタグテーブルと、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、ネットワークに存在するルートノード数と同数のツリーマネージャを生成するツリーセレクタと、

宛先ごとのフレームの転送出力先を記録するフォワーディングテーブルと、

前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、

前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器

とを備えることを特徴とするノード。

【請求項16】 前記ツリーセレクタが、  
ツリーマネージャの作成もしくは削除を行うメインコントローラと、  
フレームに付加されたタグを削除するタグ削除器と、  
スパニングツリーの切り替えのための制御フレームを送信するGVRP送受信  
器と、  
フレームにタグを付加するタグ挿入器とを備えることを特徴とする請求項15  
に記載のノード。

【請求項17】 前記ツリーマネージャが、  
フレームに付加されたタグを削除するタグ削除器と、  
BPDUsを送受信するBPDUs送受信器と、  
フレームにタグを付加するタグ挿入器と、  
スパニングツリープロトコルに従ってスパニングツリーを作成するツリーコン  
トローラと、  
前記スパニングツリープロトコルで使用するパラメータを保持するツリーテー  
ブルとを備えることを特徴とする請求項15または請求項16に記載のノード。

【請求項18】 リンクの接続状況や空き帯域を含むリソース情報を計測す  
るリソースモニタを備えることを特徴とする請求項15から請求項17の何れか  
1項に記載のノード。

【請求項19】 複数のノードを接続したネットワーク上でスパニングツリ  
ーを構成するノードにおいて、  
スパニングツリーを生成するツリーマネージャが、  
スパニングツリープロトコルの種類およびバージョンにより、コスト値を調整  
するコスト操作器を備えることを特徴とするノード。

【請求項20】 前記コスト操作器は、  
障害回復処理が遅いプロトコルを利用しているリンクに対して、大きなコスト  
を割り振ることを特徴とする請求項19に記載のノード。

【請求項21】 複数のノードを接続したネットワーク上でスパニングツリ  
ーを構成するノードにおいて、

ネットワーク内の動作の遅いプロトコルを使用中のリンクのそれぞれについて、前記各リンクのコストが最大となるスパニングツリーを生成し、前記各リンクに障害が発生した際に、前記リンクのコストが最大となっているツリーを利用してフレーム転送を行うことを特徴とするノード。

【請求項22】 複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、

独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、

転送用ツリーに対応したタグを返答するタグテーブルと、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、

ネットワークに存在する動作の遅いプロトコルを使用中のリンク数と同数のツリーマネージャを生成するツリーセレクタと、

宛先ごとのフレーム転送出力先を記録するフォワーディングテーブルと、

前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、

前記タグに従って転送先のツリーマネージャを決定する分別器とを備えることを特徴とするノード。

【請求項23】 前記ツリーセレクタが、

ツリーマネージャの作成もしくは削除を行うツリーセレクタ内のメインコントローラと、

フレームに付加されたタグを削除するタグ削除器と、

制御フレームを送信するGVRP送受信器と、

フレームにタグを付加するタグ挿入器とを備えることを特徴とする請求項22に記載のノード。

【請求項24】 前記ツリーマネージャが、

フレームに付加されたタグを削除するタグ削除器と、

BPDUsを送受信するBPDUs送受信器と、

フレームにタグを付加するタグ挿入器と、

スパニングツリープロトコルにしたがってスパニングツリーを作成するツリー

コントローラと、

スパニングツリープロトコルで使用するパラメータを保持するツリーテーブルとを備えることを特徴とする請求項22または請求項23に記載のノード。

【請求項25】 リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタを備えることを特徴とする請求項22から請求項24の何れか1項に記載のノード。

【請求項26】 スパニングツリープロトコルで利用されているHELLOフレームより、短い間隔で障害検出用フレームを送受信して障害を検出する障害検出器を備えることを特徴とする請求項4から請求項25の何れか1項に記載のノード。

【請求項27】 前記フォワーディングテーブルが、ブロードキャスト出力ポートフィールドを有することを特徴とする、請求項4、15、もしくは22に記載のノード。

【請求項28】 前記フォワーディングテーブルが、予備出力ポートフィールドを有することを特徴とする、請求項4、15、もしくは22に記載のノード。

【請求項29】 スパニングツリーによって決定されるポート種別を用いて、出力先ポートを決定することを特徴とする、請求項4、15、もしくは22に記載のノード。

【請求項30】 前記スパニングツリーによって決定されるポート種別が、ルートポート及びディジグネイティッドポートの何れかであることを特徴とする請求項29に記載のノード。

【請求項31】 複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、ネットワークの構成変更前のスパニングツリーを運用したまま、構成変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替える処理を実行することを特徴とするスパニングツリー構成プログラム。

【請求項32】 前記ネットワークの構成変更が、ノードの追加又は削除や

リンクトポロジの変化であることを特徴とする請求項31に記載のスパニングツリー構成プログラム。

【請求項33】 複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、

ネットワークのコスト変更時に、現スパニングツリーを運用したまま、リンクコスト変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替える処理を実行することを特徴とするスパニングツリー構成プログラム。

【請求項34】 複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、

複数のツリーマネージャによる独立して動作する複数のスパニングツリーを生成する処理と、

転送に利用しているスパニングツリーに対応したタグを返答する処理と、

返答のあった前記タグをフレームに挿入するタグ挿入処理と、

転送に利用するツリーを決定するツリーセレクタ処理と、

宛先ごとのフレームの転送出力先を記録するフォワーディングテーブル処理と

前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送処理と、

前記タグに従って転送先のツリーマネージャを決定する分別処理

とを実行することを特徴とするスパニングツリー構成プログラム。

【請求項35】 前記ツリーセレクタ処理において、

転送に利用するスパニングツリーの切り替えを行うコントローラ処理と、

スパニングツリーの安定を示す指定した時間のタイマ満了を知らせる安定タイマ処理と、

フレームに付加されたタグを削除するタグ削除処理と、

スパニングツリーの切り替えのための制御フレームを送信するGVRP送受信処理と、

フレームにタグを付加するタグ挿入処理

とを実行することを特徴とする請求項34に記載のスパンギングツリー構成プログラム。

【請求項36】 前記ツリーセレクタ処理において、  
スパンギングツリーの安定を示すフレームの到着間隔を判別するため、一定時間が経過すると、タイマ満了通知を送信する到着間隔タイマ処理を実行することを特徴とする請求項35に記載のスパンギングツリー構成プログラム。

【請求項37】 前記ツリーセレクタ処理は、  
リンクコストの算出に利用する指定時間のタイマ満了を知らせるコスト参照タイマ処理を実行することを特徴とする請求項34から請求項36に記載のスパンギングツリー構成プログラム。

【請求項38】 前記ツリーマネージャ処理が、  
フレームに付加されたタグを削除するタグ削除処理と、  
BPDUsを送受信するBPDUs送受信処理と、  
フレームにタグを付加するタグ挿入処理と、  
スパンギングツリープロトコルに従ってスパンギングツリーを作成するツリーコントローラ処理と、  
前記スパンギングツリープロトコルで使用するパラメータを保持するツリーテーブル処理とを実行することを特徴とする請求項34から請求項37の何れか1項に記載のスパンギングツリー構成プログラム。

【請求項39】 前記ツリーマネージャ処理は、  
通知されたリンクコストに所定の設定値を加算して返答するコスト操作処理を実行することを特徴とする請求項38に記載のスパンギングツリー構成プログラム。

【請求項40】 リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタ処理を実行することを特徴とする請求項34から請求項39の何れか1項に記載のスパンギングツリー構成プログラム。

【請求項41】 リンクコストを、利用状況により計算する処理を実行することを特徴とする請求項33に記載のスパンギングツリー構成プログラム。

【請求項42】 前記利用状況を、空き帯域と定義することを特徴とする請

求項4.1に記載のスパニングツリー構成プログラム。

【請求項4.3】 前記利用状況を、CPU負荷と定義することを特徴とする  
請求項4.1に記載のスパニングツリー構成プログラム。

【請求項4.4】 複数のノードを接続したネットワーク上でスパニングツリー  
を構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、  
ネットワーク内の各ノードがルートノードとなるスパニングツリーを生成し、  
宛先がルートノードとなるツリーを用いてフレームを転送する  
ことを特徴とするスパニングツリー構成プログラム。

【請求項4.5】 複数のノードを接続したネットワーク上でスパニングツリー  
を構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、  
独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャ  
処理と、

転送に利用しているスパニングツリーに対応したタグを返答するタグテーブル  
処理と、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入処理と  
、  
ネットワークに存在するルートノード数と同数のツリーマネージャを生成する  
ツリーセレクタ処理と、  
宛先ごとのフレーム転送出力先を記録するフォワーディングテーブル処理と、  
前記フォワーディングテーブルで指定された転送出力先にフレームを転送する  
フレーム転送処理と、  
前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別処  
理  
とを備えることを特徴とするスパニングツリー構成プログラム。

【請求項4.6】 前記ツリーセレクタ処理が、  
ツリーマネージャの作成もしくは削除を行うツリーセレクタ内のメインコント  
ローラ処理と、  
フレームに付加されたタグを削除するタグ削除処理と、  
スパニングツリーの切り替えのための制御フレームを送信するGVRP送受信

処理と、

フレームにタグを付加するタグ挿入処理

とを実行することを特徴とする請求項45に記載のスパニングツリー構成プログラム。

【請求項47】 前記ツリーマネージャ処理が、

フレームに付加されたタグを削除するタグ削除処理と、

BPDUsを送受信するBPDUs送受信処理と、

フレームにタグを付加するタグ挿入処理と、

スパニングツリープロトコルに従ってスパニングツリーを作成するツリーコントローラ処理と、

前記スパニングツリープロトコルで使用するパラメータを保持するツリーテーブル処理とを実行することを特徴とする請求項45または請求項46に記載のスパニングツリー構成プログラム。

【請求項48】 各前記ノードは、リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタ処理を実行することを特徴とする請求項45から請求項47の何れか1項に記載のスパニングツリー構成プログラム。

【請求項49】 複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノードで動作するスパニングツリー構成プログラムにおいて、

ネットワーク内の動作の遅いプロトコルを使用中のリンクのそれぞれについて、前記各リンクのコストが最大となるスパニングツリーを生成し、前記各リンクに障害が発生した際に、前記リンクのコストが最大となっているツリーを利用してフレーム転送を行うことを特徴とするスパニングツリー構成プログラム。

【請求項50】 複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノードで動作するスパニングツリー構成プログラムにおいて、

独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャ処理と、

転送用ツリーに対応したタグを返答するタグテーブル処理と、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入処理と

ネットワークに存在する動作の遅いプロトコルを使用中のリンク数と同数のツリーマネージャを生成するツリーセレクタ処理と、

宛先ごとのフレームの転送出力先を記録するフォワーディングテーブル処理と

、  
前記フォワーディングテーブルで指定された転送出力先にフレームを転送する  
フレーム転送処理と、

前記タグに従ってフレームの転送先のツリーマネージャを決定する分別処理  
とを備えることを特徴とするスパニングツリー構成プログラム。

【請求項51】 前記ツリーセレクタ処理が、

ツリーマネージャの作成もしくは削除を行うツリーセレクタ内のメインコント  
ローラ処理と、

フレームに付加されたタグを削除するタグ削除処理と、

制御フレームを送信するGVRP送受信処理と、

フレームにタグを付加するタグ挿入処理

とを備えることを特徴とする請求項50に記載のスパニングツリー構成プログ  
ラム。

【請求項52】 前記ツリーマネージャ処理が、

フレームに付加されたタグを削除するタグ削除処理と、

BPDUsを送受信するBPDUs送受信処理と、

フレームにタグを付加するタグ挿入処理と、

スパニングツリープロトコルにしたがってスパニングツリーを作成するツリー  
コントローラ処理と、

スパニングツリープロトコルで使用するパラメータを保持するツリーテーブル  
処理とを備えることを特徴とする請求項50または請求項51に記載のスパニ  
ングツリー構成プログラム。

【請求項53】 各前記ノードは、リンクの接続状況や空き帯域を含むリソ  
ース情報を計測するリソースモニタ処理を実行することを特徴とする請求項50  
から請求項52の何れか1項に記載のスパニングツリー構成プログラム。

【請求項54】 前記フォワーディングテーブルが、ブロードキャスト出力

ポートフィールドを有することを特徴とする、請求項34、45、もしくは50に記載のスパニングツリー構成プログラム。

【請求項55】 前記フォワーディングテーブルが、予備出力ポートフィールドを有することを特徴とする、請求項34、45、もしくは50に記載のスパニングツリー構成プログラム。

【請求項56】 スパニングツリーによって決定されるポート種別を用いて、出力先ポートを決定することを特徴とする、請求項34、45、もしくは50に記載のスパニングツリー構成プログラム。

【請求項57】 前記スパニングツリーによって決定されるポート種別が、ルートポート及びディジグネイティッドポートの何れかであることを特徴とする請求項56に記載のスパニングツリー構成プログラム。

【請求項58】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

各前記ノードが、

ネットワークの構成変更前のスパニングツリーを運用したまま、構成変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とするネットワークシステム。

【請求項59】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

各前記ノードが、

ネットワークのコスト変更時に、現スパニングツリーを運用したまま、リンクコスト変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とするネットワークシステム。

【請求項60】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

各前記ノードが、

独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャ

と、

転送に利用しているスパニングツリーに対応したタグを返答するタグテーブルと、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、

転送に利用するスパニングツリーを決定するツリーセレクタと、

宛先ごとのフレームの転送出力先を記録するフォワーディングテーブルと、

前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、

前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とするネットワークシステム。

【請求項61】 リンクコストを、利用状況により計算することを特徴とする請求項53に記載のネットワークシステム。

【請求項62】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

ネットワーク内の各ノードがルートノードとなるスパニングツリーを生成し、宛先がルートノードとなるツリーを用いてフレームを転送することを特徴とするネットワークシステム。

【請求項63】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、

転送用ツリーに対応したタグを返答するタグテーブルと、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、

ネットワークに存在するノード数と同数のツリーマネージャを生成するツリーセレクタと、

宛先ごとのフレーム転送出力先を記録するフォワーディングテーブルと、

前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、

前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器

とを備えることを特徴とするネットワークシステム。

【請求項64】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

スパニングツリーを生成するツリーマネージャが、

スパニングツリープロトコルの種類およびバージョンにより、コスト値を調整するコスト操作処理

を実行することを特徴とするネットワークシステム。

【請求項65】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

スパニングツリーを生成するツリーマネージャが、

スパニングツリープロトコルの種類およびバージョンにより、コスト値を調整するコスト操作器

を備えることを特徴とするネットワークシステム。

【請求項66】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

ネットワーク内の動作の遅いプロトコルを使用中のリンクのそれぞれについて、前記各リンクのコストが最大となるスパニングツリーを生成し、前記各リンクに障害が発生した際に、前記リンクのコストが最大となっているツリーを利用してフレーム転送を行う

ことを特徴とするネットワークシステム。

【請求項67】 複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、

独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、

転送用ツリーに対応したタグを返答するタグテーブルと、

前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、

ネットワークに存在する動作の遅いプロトコルを使用中のリンク数と同数のツリーマネージャを生成するツリーセレクタと、

宛先ごとのフレーム転送出力先を記録するフォワーディングテーブルと、

前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、

前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とするネットワークシステム。

【請求項68】 前記フォワーディングテーブルが、ブロードキャスト出力ポートフィールドを有することを特徴とする、請求項60、63、もしくは67に記載のネットワークシステム。

【請求項69】 前記フォワーディングテーブルが、予備出力ポートフィールドを有することを特徴とする、請求項60、63、もしくは67に記載のネットワークシステム。

【請求項70】 スパニングツリーによって決定されるポート種別を用いて、出力先ポートを決定することを特徴とする、請求項60、63、もしくは67に記載のネットワークシステム。

【請求項71】 前記スパニングツリーによって決定されるポート種別が、ルートポート及びディジグネイティッドポートの何れかであることを特徴とする請求項70に記載のネットワークシステム。

【請求項72】 複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、

ネットワークの構成変更前のスパニングツリーを運用したまま、構成変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とするスパニングツリー構成方法。

【請求項73】 複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、

ネットワークのコスト変更時に、現スパニングツリーを運用したまま、リンクコスト変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とするスパニングツリー構成方法。

【請求項74】 複数のノードを接続したネットワークにおけるスパニング

ツリー構成方法であって、

新規ノードを追加する際に、新規ノードを既存スパニングツリーに参加させず、予備スパニングツリーにのみ参加させることを特徴とするスパニングツリー構成方法。

【請求項75】 複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、

ノードを削除する際に、削除ノードを予備スパニングツリーに参加させず、既存スパニングツリーのみに参加させることを特徴とするスパニングツリー構成方法。

【請求項76】 複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、

ネットワーク構成が変更された時、予備系を用いて変更後のツリー作成を行うことを特徴とするスパニングツリー構成方法。

【請求項77】 複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、

コスト算出に、リンクの空き帯域を利用するなどを特徴とするスパニングツリー構成方法。

【請求項78】 複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、

ネットワーク内の全ノードが、全ノードをメンバに持つスパニングツリーのうち、どれか1つのルートノードとなるよう、複数のスパニングツリーを作成することを特徴とするスパニングツリー構成方法。

【請求項79】 複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、

ネットワーク内に存在する全ノードをメンバに持つスパニングツリーを作成し、うち、障害復旧の遅いプロトコルを使用しているリンクごとに複数のスパニングツリーを作成することを特徴とするスパニングツリー構成方法。

【請求項80】 複数のノードを接続したネットワークにおいて信号伝送に用いる論理トポロジを形成する方法であって、

ネットワークの構成変更前の論理トポロジを用いて信号伝送を行っている状態で、ネットワークの構成変更後の論理トポロジを生成し、

前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替えることを特徴とする論理トポロジ形成方法。

【請求項 8 1】 自らが属するネットワークにおいて信号伝送に用いる論理トポロジを用いて信号伝送を行っている状態で、前記ネットワークの構成変更時に、ネットワークの構成変更後の論理トポロジを生成する手段と、

前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替える手段とを有するノード。

【請求項 8 2】 自らが属するネットワークにおいて信号伝送に用いる論理トポロジを用いて信号伝送を行っている状態で、前記ネットワークの構成変更時に、ネットワークの構成変更後の論理トポロジを生成する処理と、

前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替える処理とを実行することを特徴とするプログラム。

【請求項 8 3】 複数のノードを接続したネットワークシステムであって、ネットワークの構成変更前の論理トポロジを用いて信号伝送を行っている状態で、ネットワークの構成変更後の論理トポロジを生成し、

前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替えることを特徴とするネットワークシステム。

【請求項 8 4】 入力されるフレームが保持する宛先に関する情報と当該フレームの転送先との対応関係をスパニングツリープロトコルを用いて作成する手段と、

前記対応関係を参照し、入力されたフレームの転送先を決定する手段とを有するノード。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ネットワークシステムに関し、特にスパニングツリーの再構成時にネットワークを停止させることが無く、さらに負荷分散機能を有するネットワークシステム、スパニングツリー構成方法及びスパニングツリー構成ノードに関する。

【0002】

【従来の技術】

従来、この種のスパニングツリーは、ループ(円環)状に形成されたネットワークにおいて、データが永遠に循環するのを防止するために用いられている。

【0003】

例えば「1998年、アイトリプルイー・スタンダード・802.1ディー(IEEE Std 802.1D)」と題するIEEE発行の標準化文書では、ループ(円環)状に形成されたネットワーク内で、データが永遠に循環するのを防止するため、ノード間でBridge Protocol Data Unit(BPDU)と呼ばれる制御情報をやり取りし、物理的にループ状になっているネットワークの一部を論理的に使用不能にして、論理的にツリー状のトポロジを形成する、スパニングツリーと呼ばれる制御手法が規定されている。これを従来技術1とする。

【0004】

また、「2001年、アイトリプルイー・スタンダード・802.1ダブリュ(IEEE Std 802.1w)」と題するIEEE発行の標準化文書では、制御情報の交換方法を拡張することにより、従来技術1におけるツリー作成を高速化し、さらに、あらかじめ迂回経路を設定しておくことにより、障害発生時の高速な迂回経路の設定を行う、高速スパニングツリーと呼ばれる制御手法が規定されている。これを従来技術2とする。

【0005】

【非特許文献1】

「1998年、アイトリプルイー・スタンダード・802.1ディー(IEEE Std 802.1D)」と題するIEEE発行の標準化文書

【非特許文献2】

「2001年、アイトリプルイー・スタンダード・802.1ダブリュ（IEEE Std 802.1w）」と題するIEEE発行の標準化文書

【0006】

【発明が解決しようとする課題】

上述した従来の技術では、以下に述べるような問題点があった。

【0007】

第1に、輻輳により、フレームの到着遅れや、欠落が発生するという問題があった。

【0008】

従来技術1では、スパニングツリーに属するノードおよびリンクの追加および削除時に、スパニングツリーを停止させて最初から構築しなおすため、再構築中に長時間ネットワーク全体が停止して輻輳が発生することにより、フレームの到着が遅れたり、欠落する場合があった。

【0009】

従来技術2では、スパニングツリーに属するノードおよびリンクの追加および削除時に、局所的にデータフレームの転送を停止させながら、徐々にスパニングツリーを構築しなおすため、再構築中にネットワークの一部が停止して輻輳し、フレームの到着が遅れたり、欠落する場合があった。

【0010】

第2に、スパニングツリーに属するノードの追加および削除等のスパニングツリー再構成時に、ネットワークが停止するという問題点があった。

【0011】

従来技術1では、スパニングツリーに属するノードの追加および削除時に、スパニングツリーを停止させて最初から構築しなおすため、再構築中に長時間ネットワーク全体が停止する場合があった。

【0012】

従来技術2では、スパニングツリーに属するノードの追加および削除時に、局所的にデータフレームの転送を停止させながら、徐々にスパニングツリーを構築しなおすため、再構築中にネットワークの一部が停止する場合があった。

【0013】

第3に、トラフィックの負荷分散ができないという問題点があった。

【0014】

従来技術1および従来技術2では、リンク容量を用いてコストを計算し、スパンニングツリー構築時の経路選択に利用しているため、トラフィックに応じた動的な負荷分散のための経路変更ができなかった。

【0015】

第4に、負荷分散しようとすると、スパンニングツリー再構成のために、ネットワークが停止するという問題点があった。

【0016】

従来技術1では、コストをトラフィック状況に応じて動的に変化させようとすると、スパンニングツリーを一旦停止させてから構築しなおすことで経路を変更するため、再構築中に長時間ネットワーク全体が停止する場合があった。

【0017】

従来技術2では、コストをトラフィック状況に応じて動的に変化させようとすると、局所的にデータフレームの転送を停止させながら、スパンニングツリーの一部を徐々に構築しなおして経路を変更するため、再構築中にネットワークの一部が停止する場合があった。

【0018】

第5に、宛先への最低コスト経路が必ずしも選択されるとは限らないという問題があった。

【0019】

従来技術1および従来技術2では、スパンニングツリーをネットワーク上で1系統のみ設定し、各ノード毎に予め設定されたプライオリティ値およびMACアドレスより、ネットワーク上で唯一のルートノードを決定し、单一のツリーを作成するため、ツリーの末端に位置するノード同士が通信する場合は、最短経路が別にあったとしてもブロッキングされてしまい、冗長な経路を通過する場合があった。

【0020】

第6に、リンク利用率が低い一方、ルートノード付近に負荷が集中するという問題があった。

【0021】

従来技術1および従来技術2では、スパニングツリーをネットワーク上で1系統のみ設定し、各ノードに設定されたプライオリティ値およびMACアドレスより、ネットワーク上で唯一のルートノードを決定し、単一のツリーを作成するため、ツリーの末端では設置されていても利用されないリンクが出現してリンク利用率が低下した。逆に、ルートノード付近では、トライフィックが集中し、輻輳発生の可能性が高まる場合があった。

【0022】

第7に、ルートノード障害時のツリー構築に時間がかかり、その間ネットワークが停止するという問題があった。

【0023】

従来技術1では、スパニングツリーをネットワーク上で1系統のみ設定し、ルートノードを1つだけしか持たないため、ルートノードにおいて障害が発生すると、スパニングツリーを停止させて最初から構築しなおすため、再構築中に長時間ネットワーク全体が停止する場合があった。

【0024】

従来技術2では、ルートノードにおいて障害が発生すると、局所的にデータフレームの転送を停止させながら、徐々にスパニングツリーを構築しなおすため、再構築中にネットワークの一部が停止する場合があった。

【0025】

第8に、IEEE802.1Dを利用する区間は、障害時のルート切り替えが遅く、スパニングツリーの再構成にも時間がかかるという問題があった。

【0026】

従来技術1では、ツリーの構築時においてデータの交換が可能となるまで、数十秒の時間がかかる場合があるからである。

【0027】

さらに、第9に、従来技術1及び従来技術2の何れにおいても、単一のツリー

しか持たないために、トライフィックがルートノード付近に集中して輻輳し、フレームの到着が遅れたり、欠落する場合があった。

【0028】

本発明の第1の目的は、輻輳発生の確率を下げ、輻輳によるフレームの到着遅れや欠落が発生する頻度を減らすことができる、ネットワークシステム、スパンギングツリー構成方法、スパンギングツリー構成ノード、及びスパンギングツリー構成プログラムを提供することにある。

【0029】

本発明の第2の目的は、ネットワークを停止させずに、スパンギングツリーに属するノードの追加および削除等のスパンギングツリー再構成ができる、ネットワークシステム、スパンギングツリー構成方法、スパンギングツリー構成ノード、及びスパンギングツリー構成プログラムを提供することにある。

【0030】

本発明の第3の目的は、トライフィックの負荷分散ができる、ネットワークシステム、スパンギングツリー構成方法、スパンギングツリー構成ノード、及びスパンギングツリー構成プログラムを提供することにある。

【0031】

本発明の第4の目的は、経路変更にともなうスパンギングツリー再構成のために、ネットワークを停止させることなく負荷分散ができる、ネットワークシステム、スパンギングツリー構成方法、スパンギングツリー構成ノード、及びスパンギングツリー構成プログラムを提供することにある。

【0032】

本発明の第5の目的は、宛先への最低コスト経路が選択される、ネットワークシステム、スパンギングツリー構成方法、スパンギングツリー構成ノード、及びスパンギングツリー構成プログラムを提供することにある。

【0033】

本発明の第6の目的は、リンク利用率を高め、ルートノード付近に負荷を集中させずに分散可能な、ネットワークシステム、スパンギングツリー構成方法、スパンギングツリー構成ノード、及びスパンギングツリー構成プログラムを提供すること

にある。

【0034】

本発明の第7の目的は、ルートノード障害によるネットワーク停止を回避可能な、ネットワークシステム、スパニングツリー構成方法、スパニングツリー構成ノード、及びスパニングツリー構成プログラムを提供することにある。

【0035】

本発明の第8の目的は、スパニングツリーがIEEE802.1D利用区間を通過して設定されるのを防ぎ、障害時の切替やルート変更を高速化し、輻輳発生の可能性やフレームの欠落の可能性を低下させることができ、ネットワークシステム、スパニングツリー構成方法、スパニングツリー構成ノード、及びスパニングツリー構成プログラムを提供することにある。

【0036】

【課題を解決するための手段】

請求項1の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、ネットワークの構成変更前のスパニングツリーを運用したまま、構成変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とする。

【0037】

請求項2の本発明のノードは、前記ネットワークの構成変更が、ノードの追加又は削除やリンクトポロジの変化であることを特徴とする。

【0038】

請求項3の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、ネットワークのコスト変更時に、現スパニングツリーを運用したまま、リンクコスト変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とする。

【0039】

請求項4の本発明は、複数のノードを接続したネットワーク上でスパニングツ

リーを構成するノードにおいて、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、転送に利用しているスパニングツリーに対応したタグを返答するタグテーブルと、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、転送に利用するスパニングツリーを決定するツリーセレクタと、宛先ごとのフレームの転送出力先を記録するフォワーディングテーブルと、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とする。

## 【0040】

請求項5の本発明のノードは、前記ツリーセレクタが、転送に利用するスパニングツリーの切り替えを行うメインコントローラと、スパニングツリーの安定を示す指定した時間のタイマ満了を知らせる安定タイマと、フレームに付加されたタグを削除するタグ削除器と、スパニングツリーの切り替えのための制御フレームを送信するGVRP送受信器と、フレームにタグを付加するタグ挿入器とを備えることを特徴とする。

## 【0041】

請求項6の本発明のノードは、前記ツリーセレクタは、スパニングツリーの安定を示すフレームの到着間隔を判別するため、一定時間が経過するとタイマ満了通知を送信する到着間隔タイマを備えることを特徴とする。

## 【0042】

請求項7の本発明のノードは、前記ツリーセレクタは、リンクコストの算出に利用する指定時間のタイマ満了を知らせるコスト参照タイマを備えることを特徴とする。

## 【0043】

請求項8の本発明のノードは、前記ツリーマネージャが、フレームに付加されたタグを削除するタグ削除器と、BPDUsを送受信するBPDUs送受信器と、フレームにタグを付加するタグ挿入器と、スパニングツリープロトコルに従ってスパニングツリーを作成するツリーコントローラと、前記スパニングツリープロトコルで使用するパラメータを保持するツリーテーブルとを備えることを特徴とする。

る。

【0044】

請求項9の本発明のノードは、前記ツリーマネージャは、通知されたリンクコストに所定の設定値を加算して返答するコスト操作器を備えることを特徴とする。

【0045】

請求項10の本発明のノードは、リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタを備えることを特徴とする。

【0046】

請求項11の本発明のノードは、前記リンクコストを、利用状況に基づいて計算することを特徴とする。

【0047】

請求項12の本発明のノードは、前記利用状況を、空き帯域と定義することを特徴とする。

【0048】

請求項13の本発明のノードは、前記利用状況を、C P U負荷と定義することを特徴とする。

【0049】

請求項14の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、ネットワーク内の各ノードがルートノードとなるスパニングツリーを生成し、宛先がルートノードとなるスパニングツリーを用いてフレームを転送することを特徴とする。

【0050】

請求項15の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、転送に利用しているスパニングツリーに対応したタグを返答するタグテーブルと、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、ネットワークに存在するルートノード数と同数のツリーマネージャを生成するツリーセレクタと、宛先ごとのフレームの転

送出力先を記録するフォワーディングテーブルと、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とする。

#### 【0051】

請求項16の本発明のノードは、前記ツリーセレクタが、ツリーマネージャの作成もしくは削除を行うメインコントローラと、フレームに付加されたタグを削除するタグ削除器と、スパニングツリーの切り替えのための制御フレームを送信するGVRP送受信器と、フレームにタグを付加するタグ挿入器とを備えることを特徴とする。

#### 【0052】

請求項17の本発明のノードは、前記ツリーマネージャが、フレームに付加されたタグを削除するタグ削除器と、BPDUsを送受信するBPDUs送受信器と、フレームにタグを付加するタグ挿入器と、スパニングツリープロトコルに従ってスパニングツリーを作成するツリーコントローラと、前記スパニングツリープロトコルで使用するパラメータを保持するツリーテーブルとを備えることを特徴とする。

#### 【0053】

請求項18の本発明のノードは、リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタを備えることを特徴とする。

#### 【0054】

請求項19の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、スパニングツリーを生成するツリーマネージャが、スパニングツリープロトコルの種類およびバージョンにより、コスト値を調整するコスト操作器を備えることを特徴とする。

#### 【0055】

請求項20の本発明のノードは、前記コスト操作器は、障害回復処理が遅いプロトコルを利用しているリンクに対して、大きなコストを割り振ることを特徴とする。

## 【0056】

請求項21の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、ネットワーク内の動作の遅いプロトコルを使用中のリンクのそれぞれについて、前記各リンクのコストが最大となるスパニングツリーを生成し、前記各リンクに障害が発生した際に、前記リンクのコストが最大となっているツリーを利用してフレーム転送を行うことを特徴とする。

## 【0057】

請求項22の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成するノードにおいて、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、転送用ツリーに対応したタグを返答するタグテーブルと、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、ネットワークに存在する動作の遅いプロトコルを使用中のリンク数と同数のツリーマネージャを生成するツリーセレクタと、宛先ごとのフレーム転送出力先を記録するフォワーディングテーブルと、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、前記タグに従って転送先のツリーマネージャを決定する分別器とを備えることを特徴とする。

## 【0058】

請求項23の本発明のノードは、前記ツリーセレクタが、ツリーマネージャの作成もしくは削除を行うツリーセレクタ内のメインコントローラと、フレームに付加されたタグを削除するタグ削除器と、制御フレームを送信するGVRP送受信器と、フレームにタグを付加するタグ挿入器とを備えることを特徴とする。

## 【0059】

請求項24の本発明のノードは、前記ツリーマネージャが、フレームに付加されたタグを削除するタグ削除器と、BPDUsを送受信するBPDUs送受信器と、フレームにタグを付加するタグ挿入器と、スパニングツリープロトコルにしたがってスパニングツリーを作成するツリーコントローラと、スパニングツリープロトコルで使用するパラメータを保持するツリーテーブルとを備えることを特徴とする。

【0060】

請求項25の本発明のノードは、リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタを備えることを特徴とする。

【0061】

請求項26の本発明のノードは、スパンニングツリープロトコルで利用されているHELLOフレームより、短い間隔で障害検出用フレームを送受信して障害を検出する障害検出器を備えることを特徴とする。

【0062】

請求項27の本発明のノードは、前記フォワーディングテーブルが、ブロードキャスト出力ポートフィールドを有することを特徴とする。

【0063】

請求項28の本発明のノードは、前記フォワーディングテーブルが、予備出力ポートフィールドを有することを特徴とする。

【0064】

請求項29の本発明のノードは、スパンニングツリーによって決定されるポート種別を用いて、出力先ポートを決定することを特徴とする。

【0065】

請求項30の本発明のノードは、前記スパンニングツリーによって決定されるポート種別が、ルートポート及びディジグネイティッドポートの何れかであることを特徴とする。

【0066】

請求項31の本発明は、複数のノードを接続したネットワーク上でスパンニングツリーを構成する各ノード上で動作するスパンニングツリー構成プログラムにおいて、ネットワークの構成変更前のスパンニングツリーを運用したまま、構成変更後の新たなスパンニングツリーを生成し、前記新たなスパンニングツリーの安定後に、転送に利用するスパンニングツリーを前記新たなスパンニングツリーに切り替える処理を実行することを特徴とする。

【0067】

請求項32の本発明のスパンニングツリー構成プログラムは、前記ネットワーク

の構成変更が、ノードの追加又は削除やリンクトポロジの変化であることを特徴とする。

## 【0068】

請求項33の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、ネットワークのコスト変更時に、現スパニングツリーを運用したまま、リンクコスト変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替える処理を実行することを特徴とする。

## 【0069】

請求項34の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、複数のツリーマネージャによる独立して動作する複数のスパニングツリーを生成する処理と、転送に利用しているスパニングツリーに対応したタグを返答する処理と、返答のあった前記タグをフレームに挿入するタグ挿入処理と、転送に利用するツリーを決定するツリーセレクタ処理と、宛先ごとのフレームの転送出力先を記録するフォワーディングテーブル処理と、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送処理と、前記タグに従って転送先のツリーマネージャを決定する分別処理とを実行することを特徴とする。

## 【0070】

請求項35の本発明のスパニングツリー構成プログラムは、前記ツリーセレクタ処理において、転送に利用するスパニングツリーの切り替えを行うコントローラ処理と、スパニングツリーの安定を示す指定した時間のタイマ満了を知らせる安定タイマ処理と、フレームに付加されたタグを削除するタグ削除処理と、スパニングツリーの切り替えのための制御フレームを送信するGVRP送受信処理と、フレームにタグを付加するタグ挿入処理とを実行することを特徴とする。

## 【0071】

請求項36の本発明のスパニングツリー構成プログラムは、前記ツリーセレク

タ処理において、スパンギングツリーの安定を示すフレームの到着間隔を判別するため、一定時間が経過すると、タイマ満了通知を送信する到着間隔タイマ処理を実行することを特徴とする。

【0072】

請求項37の本発明のスパンギングツリー構成プログラムは、前記ツリーセレクタ処理は、リンクコストの算出に利用する指定時間のタイマ満了を知らせるコスト参照タイマ処理を実行することを特徴とする。

【0073】

請求項38の本発明のスパンギングツリー構成プログラムは、前記ツリーマネージャ処理が、フレームに付加されたタグを削除するタグ削除処理と、BPDUsを送受信するBPDUs送受信処理と、フレームにタグを付加するタグ挿入処理と、スパンギングツリープロトコルに従ってスパンギングツリーを作成するツリーコントローラ処理と、前記スパンギングツリープロトコルで使用するパラメータを保持するツリーテーブル処理とを実行することを特徴とする。

【0074】

請求項39の本発明のスパンギングツリー構成プログラムは、前記ツリーマネージャ処理は、通知されたリンクコストに所定の設定値を加算して返答するコスト操作処理を実行することを特徴とする。

【0075】

請求項40の本発明のスパンギングツリー構成プログラムは、リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタ処理を実行することを特徴とする。

【0076】

請求項41の本発明のスパンギングツリー構成プログラムは、リンクコストを、利用状況により計算する処理を実行することを特徴とする。

【0077】

請求項42の本発明のスパンギングツリー構成プログラムは、前記利用状況を、空き帯域と定義することを特徴とする。

【0078】

請求項43の本発明のスパニングツリー構成プログラムは、前記利用状況を、C P U負荷と定義することを特徴とする。

## 【0079】

請求項44の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、ネットワーク内の各ノードがルートノードとなるスパニングツリーを生成し、宛先がルートノードとなるツリーを用いてフレームを転送することを特徴とする。

## 【0080】

請求項45の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノード上で動作するスパニングツリー構成プログラムにおいて、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャ処理と、転送に利用しているスパニングツリーに対応したタグを返答するタグテーブル処理と、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入処理と、ネットワークに存在するルートノード数と同数のツリーマネージャを生成するツリーセレクタ処理と、宛先ごとのフレーム転送出力先を記録するフォワーディングテーブル処理と、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送処理と、前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別処理とを備えることを特徴とする。

## 【0081】

請求項46の本発明のスパニングツリー構成プログラムは、前記ツリーセレクタ処理が、ツリーマネージャの作成もしくは削除を行うツリーセレクタ内のメインコントローラ処理と、フレームに付加されたタグを削除するタグ削除処理と、スパニングツリーの切り替えのための制御フレームを送信するG V R P送受信処理と、フレームにタグを付加するタグ挿入処理とを実行することを特徴とする。

## 【0082】

請求項47の本発明のスパニングツリー構成プログラムは、前記ツリーマネージャ処理が、フレームに付加されたタグを削除するタグ削除処理と、B P D Uを

送受信するBPDUsの送受信処理と、フレームにタグを付加するタグ挿入処理と、スパニングツリープロトコルに従ってスパニングツリーを作成するツリーコントローラ処理と、前記スパニングツリープロトコルで使用するパラメータを保持するツリーテーブル処理とを実行することを特徴とする。

## 【0083】

請求項48の本発明のスパニングツリー構成プログラムは、各前記ノードは、リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタ処理を実行することを特徴とする。

## 【0084】

請求項49の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノードで動作するスパニングツリー構成プログラムにおいて、ネットワーク内の動作の遅いプロトコルを使用中のリンクのそれぞれについて、前記各リンクのコストが最大となるスパニングツリーを生成し、前記各リンクに障害が発生した際に、前記リンクのコストが最大となっているツリーを利用してフレーム転送を行うことを特徴とする。

## 【0085】

請求項50の本発明は、複数のノードを接続したネットワーク上でスパニングツリーを構成する各ノードで動作するスパニングツリー構成プログラムにおいて、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャ処理と、転送用ツリーに対応したタグを返答するタグテーブル処理と、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入処理と、ネットワークに存在する動作の遅いプロトコルを使用中のリンク数と同数のツリーマネージャを生成するツリーセレクタ処理と、宛先ごとのフレームの転送出力先を記録するフォワーディングテーブル処理と、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送処理と、前記タグに従ってフレームの転送先のツリーマネージャを決定する分別処理とを備えることを特徴とする。

## 【0086】

請求項51の本発明のスパニングツリー構成プログラムは、前記ツリーセレク

タ処理が、ツリーマネージャの作成もしくは削除を行うツリーセレクタ内のメインコントローラ処理と、フレームに付加されたタグを削除するタグ削除処理と、制御フレームを送信するGVRP送受信処理と、フレームにタグを付加するタグ挿入処理とを備えることを特徴とする。

## 【0087】

請求項52の本発明のスパンニングツリー構成プログラムは、前記ツリーマネージャ処理が、フレームに付加されたタグを削除するタグ削除処理と、BPDUを送受信するBPDU送受信処理と、フレームにタグを付加するタグ挿入処理と、スパンニングツリープロトコルにしたがってスパンニングツリーを作成するツリーコントローラ処理と、スパンニングツリープロトコルで使用するパラメータを保持するツリーテーブル処理とを備えることを特徴とする。

## 【0088】

請求項53の本発明のスパンニングツリー構成プログラムは、各前記ノードは、リンクの接続状況や空き帯域を含むリソース情報を計測するリソースモニタ処理を実行することを特徴とする。

## 【0089】

請求項54の本発明のスパンニングツリー構成プログラムは、前記フォワーディングテーブルが、ブロードキャスト出力ポートフィールドを有することを特徴とする。

## 【0090】

請求項55の本発明のスパンニングツリー構成プログラムは、前記フォワーディングテーブルが、予備出力ポートフィールドを有することを特徴とする。

## 【0091】

請求項56の本発明のスパンニングツリー構成プログラムは、スパンニングツリーによって決定されるポート種別を用いて、出力先ポートを決定することを特徴とする。

## 【0092】

請求項57の本発明のスパンニングツリー構成プログラムは、前記スパンニングツリーによって決定されるポート種別が、ルートポート及びディジグネイティッドポ

ートの何れかであることを特徴とする。

【0093】

請求項58の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、各前記ノードが、ネットワークの構成変更前のスパニングツリーを運用したまま、構成変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とする。

【0094】

請求項59の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、各前記ノードが、ネットワークのコスト変更時に、現スパニングツリーを運用したまま、リンクコスト変更後の新たなスパニングツリーを生成し、前記新たなスパニングツリーの安定後に、転送に利用するスパニングツリーを前記新たなスパニングツリーに切り替えることを特徴とする。

【0095】

請求項60の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、各前記ノードが、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、転送に利用しているスパニングツリーに対応したタグを返答するタグテーブルと、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、転送に利用するスパニングツリーを決定するツリーセレクタと、宛先ごとのフレームの転送出力先を記録するフォワーディングテーブルと、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とする。

【0096】

請求項61の本発明のネットワークシステムは、リンクコストを、利用状況により計算することを特徴とする。

## 【0097】

請求項62の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、ネットワーク内の各ノードがルートノードとなるスパニングツリーを生成し、宛先がルートノードとなるツリーを用いてフレームを転送することを特徴とする。

## 【0098】

請求項63の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、転送用ツリーに対応したタグを返答するタグテーブルと、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、ネットワークに存在するノード数と同数のツリーマネージャを生成するツリーセレクタと、宛先ごとのフレーム転送出力先を記録するフォワーディングテーブルと、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とする。

## 【0099】

請求項64の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、スパニングツリーを生成するツリーマネージャが、スパニングツリープロトコルの種類およびバージョンにより、コスト値を調整するコスト操作処理を実行することを特徴とする。

## 【0100】

請求項65の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、スパニングツリーを生成するツリーマネージャが、スパニングツリープロトコルの種類およびバージョンにより、コスト値を調整するコスト操作器を備えることを特徴とする。

## 【0101】

請求項66の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、ネットワーク内の動作の遅いプロトコルを使用中のリンクのそれぞれについて、前記各リンクのコストが最大となるスパニングツリーを生成し、前記各リンクに障害が発生した際に、前記リンクのコストが最大となっているツリーを利用してフレーム転送を行うことを特徴とする。

## 【0102】

請求項67の本発明は、複数のノードを接続したネットワーク上でスパニングツリーによって転送経路を設定するネットワークシステムにおいて、独立して動作する複数のスパニングツリーを生成する複数のツリーマネージャと、転送用ツリーに対応したタグを返答するタグテーブルと、前記タグテーブルより返答のあったタグをフレームに挿入するタグ挿入器と、ネットワークに存在する動作の遅いプロトコルを使用中のリンク数と同数のツリーマネージャを生成するツリーセレクタと、宛先ごとのフレーム転送出力先を記録するフォワーディングテーブルと、前記フォワーディングテーブルで指定された転送出力先にフレームを転送するフレーム転送器と、前記タグに従って前記フレームの転送先のツリーマネージャを決定する分別器とを備えることを特徴とする。

## 【0103】

請求項68の本発明のネットワークシステムは、前記フォワーディングテーブルが、ブロードキャスト出力ポートフィールドを有することを特徴とする。

## 【0104】

請求項69の本発明のネットワークシステムは、前記フォワーディングテーブルが、予備出力ポートフィールドを有することを特徴とする。

## 【0105】

請求項70の本発明のネットワークシステムは、スパニングツリーによって決定されるポート種別を用いて、出力先ポートを決定することを特徴とする。

## 【0106】

請求項71の本発明のネットワークシステムは、前記スパニングツリーによって決定されるポート種別が、ルートポート及びディジグネイテッドポートの何れ

かであることを特徴とする。

#### 【0107】

請求項72の本発明は、複数のノードを接続したネットワークにおけるスパンギツリー構成方法であって、ネットワークの構成変更前のスパンギツリーを運用したまま、構成変更後の新たなスパンギツリーを生成し、前記新たなスパンギツリーの安定後に、転送に利用するスパンギツリーを前記新たなスパンギツリーに切り替えることを特徴とする。

#### 【0108】

請求項73の本発明は、複数のノードを接続したネットワークにおけるスパンギツリー構成方法であって、ネットワークのコスト変更時に、現スパンギツリーを運用したまま、リンクコスト変更後の新たなスパンギツリーを生成し、前記新たなスパンギツリーの安定後に、転送に利用するスパンギツリーを前記新たなスパンギツリーに切り替えることを特徴とする。

#### 【0109】

請求項74の本発明は、複数のノードを接続したネットワークにおけるスパンギツリー構成方法であって、新規ノードを追加する際に、新規ノードを既存スパンギツリーに参加させず、予備スパンギツリーにのみ参加させることを特徴とする。

#### 【0110】

請求項75の本発明は、複数のノードを接続したネットワークにおけるスパンギツリー構成方法であって、ノードを削除する際に、削除ノードを予備スパンギツリーに参加させず、既存スパンギツリーのみに参加させることを特徴とする。

#### 【0111】

請求項76の本発明は、複数のノードを接続したネットワークにおけるスパンギツリー構成方法であって、ネットワーク構成が変更された時、予備系を用いて変更後のツリー作成を行うことを特徴とする。

#### 【0112】

請求項77の本発明は、複数のノードを接続したネットワークにおけるスパン

ングツリー構成方法であって、コスト算出に、リンクの空き帯域を利用することを特徴とする。

【0113】

請求項78の本発明は、複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、ネットワーク内の全ノードが、全ノードをメンバに持つスパニングツリーのうち、どれか1つのルートノードとなるよう、複数のスパニングツリーを作成することを特徴とする。

【0114】

請求項79の本発明は、複数のノードを接続したネットワークにおけるスパニングツリー構成方法であって、ネットワーク内に存在する全ノードをメンバに持つスパニングツリーを作成し、うち、障害復旧の遅いプロトコルを使用しているリンクごとに複数のスパニングツリーを作成することを特徴とする。

【0115】

請求項80の本発明は、複数のノードを接続したネットワークにおいて信号伝送に用いる論理トポロジを形成する方法であって、ネットワークの構成変更前の論理トポロジを用いて信号伝送を行っている状態で、ネットワークの構成変更後の論理トポロジを生成し、前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替えることを特徴とする。

【0116】

請求項81の本発明のノードは、自らが属するネットワークにおいて信号伝送に用いる論理トポロジを用いて信号伝送を行っている状態で、前記ネットワークの構成変更時に、ネットワークの構成変更後の論理トポロジを生成する手段と、前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替える手段とを有する。

【0117】

請求項82の本発明のプログラムは、自らが属するネットワークにおいて信号伝送に用いる論理トポロジを用いて信号伝送を行っている状態で、前記ネットワークの構成変更時に、ネットワークの構成変更後の論理トポロジを生成する処理

と、前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替える処理とを実行することを特徴とする。

## 【0118】

請求項83の本発明は、複数のノードを接続したネットワークシステムであつて、ネットワークの構成変更前の論理トポロジを用いて信号伝送を行っている状態で、ネットワークの構成変更後の論理トポロジを生成し、前記構成変更後の論理トポロジ安定後に、信号伝送に用いる論理トポロジを前記構成変更後の論理トポロジに切り替えることを特徴とする。

## 【0119】

請求項84の本発明のノードは、入力されるフレームが保持する宛先に関する情報と当該フレームの転送先との対応関係をスパニングツリープロトコルを用いて作成する手段と、前記対応関係を参照し、入力されたフレームの転送先を決定する手段とを有する。

## 【0120】

## 【発明の実施の形態】

以降の説明の中で、複数のスパニングツリーおよび複数のノード群を識別する識別子としてタグを用いた説明を行うが、このタグは、VLANタグのほかに、本特許出願人による特願2002-204673号に開示される拡張タグ及び他のタグもしくは識別手段のうち、単独もしくは何れか1つ以上の組み合わせを意味する。

## 【0121】

ここで、本発明において用いるタグのうち、上記特願2002-204673号に開示される拡張タグ付きフレームのフォーマットについて説明する。

## 【0122】

図1は、IEEE802.1Qで規定されているVLANタグ付きのイーサネット(R)フレームのフォーマットである。VLANタグ付きイーサネット(R)フレーム3200は、送信先MACアドレス3201と、送信元MACアドレス3202と、VLANタグ3203と、イーサネット(R)属性情報3204と、ペイ

ロード3205と、FCS3206とから構成される。

#### 【0123】

これに対して、図2は、本発明の拡張タグ付きイーサネット(R)フレームのフォーマットである。拡張タグ付きイーサネット(R)フレーム3300は、送信先MACアドレス3201と、送信元MACアドレス3202と、拡張タグ格納領域3301と、イーサネット(R)属性情報3204と、ペイロード3205と、FCS3206とから構成され、既存のVLANタグ付きイーサネット(R)フレーム3200のVLANタグ3203が拡張タグ格納領域3301に置き換わる。

#### 【0124】

また図3に示すように別の構成の拡張タグ付きイーサネット(R)フレーム3400も存在し、これは送信先MACアドレス3201と、送信元MACアドレス3202と、拡張タグ格納領域3301と、VLANタグ3203と、イーサネット(R)属性情報3204と、ペイロード3205と、FCS3206とから構成され、拡張タグ格納領域3301は送信元MACアドレス3202の後に挿入される。

#### 【0125】

拡張タグ格納領域3301には1つまたは複数の拡張タグを格納可能である。拡張タグのサイズは4バイトであり、VLANタグ3203と同一サイズとなっている。拡張タグ付きイーサネット(R)フレーム3300、3400の最上段の拡張タグとVLANタグ付きイーサネット(R)フレーム3200のVLANタグは同一の位置に同一のサイズで格納されており、その区別は各々のタグの上位2バイトに格納される値を変更することにより区別する（詳細については後述する）。

#### 【0126】

これにより、拡張タグ付きイーサネット(R)フレーム3300、3400は、VLANタグ付きイーサネット(R)フレーム3200と互換性を有しており、既存ノード、拡張タグ対応ノードの双方においてどちらのフレームも処理可能である。

## 【0127】

図4は、拡張タグ格納領域3301を示している。図4に示す格納例では、8個の拡張タグ3500～3507が格納されている。

## 【0128】

フォワーディングタグ3500には宛先ノードの識別子や宛先までのラベル（例えばMPLSラベル）が格納される。また、宛先ノードの識別子を格納したフォワーディングタグ3500に加えて、送信元ノードの識別子を格納する場合もある。各ノードはこのフォワーディングタグを参照してフレーム転送先を決定する。このフォワーディングタグ3500は拡張タグ付きイーサネット(R)フレーム3300、3400において必ず格納される。

## 【0129】

拡張タグの種類としては、カスタマ分離タグ3501、プロテクションタグ3502、OAM&Pタグ3503、品質情報タグ3504、フレーム制御タグ3505、セキュリティタグ3506、ユーザ拡張タグ3507が格納されている。

## 【0130】

カスタマ分離タグ3502には、各ノードに収容されるカスタマ毎に情報を分離するための識別子が格納される。カスタマとしては、同一のVLANの所属するカスタマを同一カスタマとする場合や、2つ以上のノードの特定のポートに収容されるカスタマを同一カスタマとする場合や、網内のノードに接続する2つ以上のホストを同一カスタマとする場合などがある。これらのカスタマに対して分離識別子が割り当てられ、各カスタマからのフレームにはカスタマ分離タグ3502内にこの分離識別子が格納される。カスタマ分離タグ3502によってカスタマを識別することにより、カスタマ単位の付加サービス（例えば、特定カスタマに対する優先制御など）の提供が可能となる。また、カスタマ分離タグ3502は複数スタックして使うことも可能である。この場合、分離可能なカスタマ数を大幅に増やすことができる。なお、カスタマ分離タグ3502をスタックする場合には、スタックした最終段のカスタマ分離タグ3502は最終段であることを示す特別なカスタマ分離タグを使用する。

## 【0131】

プロテクションタグ3503には、障害発生時の障害情報や障害復旧のための迂回経路情報が格納される。OAM&Pタグ3504には、運用／管理情報が格納される。

## 【0132】

品質情報タグ3505には、遅延、ジッタ、パケットロス率やフレームのネットワーク内への流入時間を示すタイムスタンプ、帯域制御情報等の品質情報が格納される。品質情報タグ3505にタイムスタンプ値が格納されている場合、フレームを受信したノードは現在の時刻とタイムスタンプ値よりそのフレームの網内遅延（ネットワーク内での滞在時間）を算出することができる。網内遅延の保証値が定められている場合には、保証値を実現できるよう優先処理を実施することができる。また、品質情報タグ3505に要求帯域や蓄積データ量やトラヒッククラス等の帯域制御情報が格納されている場合、そのフローの蓄積データ量やトラヒッククラスと他のフローのトラヒック状況を考慮して、要求帯域を確保するための帯域制御を実施することができる。

## 【0133】

フレーム制御タグ3506には、フレームのネットワーク内での生存時間を制限するホップカウンタ（TTL : Time To Live）や誤り検出のためのCRCなどの情報が格納される。TTLが格納される場合には、経由するノード毎にTTL値が減算され、TTL=0となったところでこのフレームは廃棄される。これにより、ループ経路になった場合でもフレームが循環し続けるのを防ぐことができる。またCRCが格納される場合には、入口側ノードにおける拡張タグ格納領域3301のCRC演算結果が格納されており、出口側ノードにおいて再度CRC演算を実施して格納値と比較することにより、拡張タグ格納領域3301の誤りを検出可能である。

## 【0134】

セキュリティタグ3507には、フレームの信頼性、ネットワーク構築時やネットワーク構成変更時の秘匿性を確保するための情報が格納される。セキュリティタグ3507の利用例としては以下の例があげられる。ネットワーク内で通信

するカスタマにはカスタマ毎のセキュリティ識別子が予め設定され、その識別子はカスタマが接続する各ノードで保持される。各カスタマはフレーム転送の際に、設定されたセキュリティ識別子を常にセキュリティタグ3507に格納することにより、カスタマ分離タグ3501の情報を改ざんした悪意のあるカスタマからのフレーム送受信を防ぐことができる。また、ネットワーク構築時やネットワーク構成変更時において、ノード間でネゴシエートして共通のセキュリティ識別子を設定する。当該ノード間のフレーム転送の際に、設定したセキュリティ識別子を常にセキュリティタグ3507に格納することにより、悪意のあるノードをネットワークに接続するのを防ぐことができる。

#### 【0135】

ユーザ拡張タグ3507には、ユーザが独自に定義する任意の情報が格納される。ユーザが独自にタグのフォーマット及び格納情報を定義し、その処理内容を定義することにより、ユーザ独自の機能拡張を図ることができ、ネットワークの柔軟性を高められる。

#### 【0136】

フォワーディングタグ3500以外の拡張タグ3501～3507は、必要に応じて格納される。フォワーディングタグ3500は、拡張タグ格納領域3301の先頭に格納され、その他の拡張タグ3501～3507はその後方に格納される。フォワーディングタグ3500よりも後方であれば、予め決められた固定位置でも、任意の位置でも良い。

#### 【0137】

以降、2系統存在するスパンギングツリーのうち、新しくネットワークに侵入するデータフレームの転送に利用しているスパンギングツリーを現用ツリーもしくは現用系ツリー、現用ツリーとなっていないツリーを予備用ツリーもしくは予備系ツリーと表現する。

#### 【0138】

また、現用系ツリーの生成を行うツリーマネージャを現用系ツリーマネージャ、予備系ツリーの生成を行うツリーマネージャを、予備系ツリーマネージャと呼ぶ。

## 【0139】

タググループとは、タグおよびその他の識別子を利用して識別されるノード群、すなわち複数のノードの集合体をいう。タググループをVLANタグを識別子として形成した場合は、前記タググループをVLANと呼ぶ。

## 【0140】

BPDUs(Bridge Protocol Data Unit)とは、スパニングツリー生成のために交換されるIEEE802.1D(従来技術1)およびIEEE802.1w(従来技術2)に記載される制御データおよび、本発明の現用系と予備系の識別情報等を含む制御フレームのことである。

## 【0141】

図5は、IEEE802.1D(従来技術1)およびIEEE802.1w(従来技術2)に記載の、Configuration BPDUsフレーム2205の構造を示すフォーマット図である。

## 【0142】

MAC DA2201は、宛先MACアドレスを格納する領域である。

## 【0143】

MAC SA2202は、送信元MACアドレスを格納する領域である。

## 【0144】

タグ領域2203は、複数のスパニングツリーを識別する識別子としてのタグを挿入する領域である。また、従来技術には記載されていないが、前記タグは、VLANタグのほかに、本特許出願人による特願2002-204673号に開示される拡張タグ、及び他のタグもしくは識別手段の何れか1つ以上の組み合わせたタグでもよい。

## 【0145】

Type2204は、フレームのタイプ識別子を格納する領域である。

## 【0146】

BPDUs領域2205は、IEEE802.1D(従来技術1)およびIEEE802.1w(従来技術2)に記載の、Configuration BPDUs parametersに相当する情報を格納する領域である。

【0147】

F C S 2 2 0 6 は、フレームチェックシーケンスを格納する領域である。

【0148】

P r o t o c o l I d e n t i f i e r 2 2 0 5 1 は、I E E E 8 0 2 . 1 D (従来技術1) もしくはI E E E 8 0 2 . 1 w (従来技術2) に記載の、P r o t o c o l I d e n t i f i e r と等しい情報を格納する領域である。

【0149】

P r o t o c o l V e r s i o n I d e n t i f i e r 2 2 0 5 2 は、I E E E 8 0 2 . 1 D (従来技術1) もしくはI E E E 8 0 2 . 1 w (従来技術2) に記載の、P r o t o c o l V e r s i o n I d e n t i f i e r と等しい情報を格納する領域である。

【0150】

B P D U T y p e 2 2 0 5 3 は、I E E E 8 0 2 . 1 D (従来技術1) もしくはI E E E 8 0 2 . 1 w (従来技術2) に記載の、B P D U T y p e と等しい情報を格納する領域である。

【0151】

F l a g s 2 2 0 5 4 は、I E E E 8 0 2 . 1 D (従来技術1) もしくはI E E E 8 0 2 . 1 w (従来技術2) に記載の、F l a g s と等しい情報を格納する領域である。

【0152】

R o o t I d e n t i f i e r 2 2 0 5 5 は、I E E E 8 0 2 . 1 D (従来技術1) もしくはI E E E 8 0 2 . 1 w (従来技術2) に記載の、R o o t I d e n t i f i e r と等しい情報を格納する領域である。

【0153】

R o o t P a t h C o s t 2 2 0 5 6 は、I E E E 8 0 2 . 1 D (従来技術1) もしくはI E E E 8 0 2 . 1 w (従来技術2) に記載の、R o o t P a t h C o s t と等しい情報を格納する領域である。

【0154】

B r i d g e I d e n t i f i e r 2 2 0 5 7 は、I E E E 8 0 2 . 1 D (

従来技術1) もしくはIEEE802.1w(従来技術2)に記載の、Bridge Identifierと等しい情報を格納する領域である。

【0155】

Port Identifier22058は、IEEE802.1D(従来技術1) もしくはIEEE802.1w(従来技術2)に記載の、Port Identifierと等しい情報を格納する領域である。

【0156】

Message Age22059は、IEEE802.1D(従来技術1) もしくはIEEE802.1w(従来技術2)に記載の、Message Ageと等しい情報を格納する領域である。

【0157】

MAX Age2205Aは、IEEE802.1D(従来技術1) もしくはIEEE802.1w(従来技術2)に記載の、MAX Ageと等しい情報を格納する領域である。

【0158】

Hello Time2205Bは、IEEE802.1D(従来技術1) もしくはIEEE802.1w(従来技術2)に記載の、Hello Timeと等しい情報を格納する領域である。

【0159】

Forward Delay2205Cは、IEEE802.1D(従来技術1) もしくはIEEE802.1w(従来技術2)に記載の、Forward Delayと等しい情報を格納する領域である。

【0160】

図6は、IEEE802.1D(従来技術1) およびIEEE802.1w(従来技術2)に記載の、Topology Change Notification BPDUフレームの構造を示すフォーマット図である。

【0161】

MAC DA2201は、宛先MACアドレスを格納する領域である。

【0162】

MAC SA2202は、送信元MACアドレスを格納する領域である。

【0163】

タグ領域2203は、従来技術には記載されていないが、複数のスパンニングツリーを識別する識別子としてのタグを挿入する領域である。前記タグは、VLANタグのほかに、上記特願2002-204673号に開示される拡張タグ、及び他のタグもしくは識別手段の何れか1つ以上の組み合わせたタグでもよい。

【0164】

Type2204は、フレームのタイプ識別子を格納する領域である。

【0165】

BPDU領域2205は、IEEE802.1D（従来技術1）およびIEEE802.1w（従来技術2）に記載の、Topology Change Notification BPDU parametersに相当する情報を格納する領域である。

【0166】

FCS2206は、フレームチェックシーケンスを格納する領域である。

【0167】

GVRPとは、タググループの管理および、現用系と予備系の識別、および、ノード間での各種設定情報の交換のために送受信される、制御フレームのことを行う。

【0168】

図53において、拡張タグフレーム3300および3400の他のフレームフォーマットについて説明する。なお、以降では図4で説明した拡張タグフレーム3500～3508のフレームフォーマットを拡張タグフレームフォーマット（1）と記し、以降で図53において説明するフレームフォーマットを拡張タグフレームフォーマット（2）と記すこととする。

【0169】

図53の上部はVLANタグ3203の詳細なフレームフォーマットである。TPID（Tag Protocol Identifier）2800には、値「0x8100」が設定されている。なお、標準規格外ではあるが値「0x9

100」が使用される場合もある。また、TCI2801はPriorityフィールド2802とCFI2803とVLAN-IDフィールド2804から構成される。

#### 【0170】

Priorityフィールド2802にはフレームの優先度が格納され、その優先度の値はIEEE802.1pにおいて規定されている。また、CFIには特殊ルーティング情報の有無またはMACアドレスのフォーマットの種別を示す値が格納され、VLAN-IDフィールド2804にはVLAN-IDが格納される。

#### 【0171】

これに対して、図50の下部に示す拡張タグフレームフォーマット(2)では、TPID2800及びTCI2801の中のCFI2803はVLANタグ3203と同様であり、Priorityフィールド2802がPriority／タグTypeフィールド5003に、VLAN-IDフィールド2804が拡張タグ情報フィールド5004に変更される。なお、対応するフィールドのサイズは同様である。

#### 【0172】

本拡張タグフレームフォーマット(2)では、Priority／タグTypeフィールド5003において、拡張タグ3500～3508の種別を格納する。拡張タグ3500～2307使用時にもIEEE802.1pをサポートできるように、既存のVLANタグ3203のPriorityフィールド2802(IEEE802.1p)におけるPriority値の一部を拡張タグ3500～3508の種別として用いる。

#### 【0173】

具体的には110、100、001、000を拡張タグ3500～3508用として用いることとし、111(予約用)、101(会話型マルチメディア用)、011(クリティカルアプリケーション用)、010(標準的ストリーム用)についてはIEEE802.1p互換とする。

#### 【0174】

そのため、使用できる拡張タグ3500～3508は4つに制限しており、例えばフォワーディングタグ3500、ブロードキャストフォワーディングタグ3508、カスタマ分離タグ3501、OAM&Pタグ3503を使用し、Priority値との対応を001=フォワーディングタグ3500、000=ブロードキャストフォワーディングタグ3508、110=カスタマ分離タグ3501、100=OAM&Pタグ3503とする。これにより、上記4種類の拡張タグを識別できると共に、IEEE802.1pの中の4つの優先度をサポートできる。なお、使用する拡張タグの選択とそれに対応するPriority値の設定についてはこの例に限定されるわけではない。

#### 【0175】

また拡張タグフレームフォーマット(2)では、拡張タグ情報フィールド5004において、拡張タグ3500～3508のタグ種別に応じたアドレス情報などの情報が格納される。例えば、フォワーディングタグ3500の場合、宛先ノードのアドレス情報が格納され、ブロードキャストフォワーディングタグ3508の場合、送信元ノードのアドレス情報が格納され、カスタマ分離タグ3501の場合、カスタマの識別情報が格納される。

#### 【0176】

##### (第1の実施の形態)

以下、本発明の第1の実施の形態について、図面を参照して詳細に説明する。

#### 【0177】

図7を参照すると、本発明の第1の実施の形態は、ノード11～16、クライアント91～96、リンク81～86、リンク21～28を含む。

#### 【0178】

ノード11は、プログラム制御されるCPU等で実現され、以下に挙げる機能を有する。

#### 【0179】

(1) リンク21もしくはリンク24より到着したフレームを、リンク24もしくはリンク21に転送する。

#### 【0180】

(2) リンク81より到着したフレームに、転送に必要なタグを付加した上で、リンク21もしくはリンク24に転送する。

【0181】

(3) リンク21もしくはリンク24より到着したフレームを、転送に必要なタグを削除した上で、リンク81に転送する。

【0182】

(4) スパニングツリーを構成するために、他のノードとの間で制御フレームのを行い、必要に応じてリンクのポートを閉鎖する。

【0183】

(5) リンクを流れるフレームの流量をモニタする。

【0184】

ノード12～16は、ノード11と同様のノードである。以後、ノード11～16を代表して、ノード11を用いて記述を行うが、これらノード11についての記述は、特に断りのない限り他のノード12～16においても同様に実現可能である。

【0185】

クライアント91は、1つ以上のクライアントの集合であり、リンク81を通じて、ノード11との間でフレームの送受信を行う機能を有する。

【0186】

クライアント92～96は、クライアント91と同様のクライアント群である。以後、クライアント91～96を代表してクライアント91を用いて記述を行うが、クライアント91についての記述は、特に断りのない限り他のクライアント92～96においても同様に適用可能である。

【0187】

リンク81は、クライアント91からノード11、及び、ノード11からクライアント91を結ぶ双方向リンクである。

【0188】

リンク82～86は、リンク81と同様のリンクである。以後、リンク81～86を代表してリンク81を用いて記述を行うが、リンク81についての記述は

、特に断りのない限り他のリンク82～86においても同様に適用可能である。

【0189】

リンク21は、ノード11からノード12、及び、ノード12からノード11を結ぶ、双向リンクである。

【0190】

リンク22～26は、リンク21と同様のリンクである。以後、リンク22～26を代表してリンク21を用いて記述を行うが、リンク21についての記述は、特に断りのない限り他のリンク22～26においても同様に適用可能である。

【0191】

図8は、ノード11の構成を詳細に示した図である。ノード11は、フレーム転送器111と、タグ挿入器112と、タグ削除器113と、フォワーディングテーブル114と、分別器1150と、ツリーマネージャ1151と、ツリーマネージャ1152と、ツリーセレクタ116と、タグテーブル117と、設定インターフェース118を含む。

【0192】

フレーム転送器111は、リンク21もしくはリンク24、およびタグ挿入器112より受信したフレームを、フォワーディングテーブル114の記述に従い、リンク21もしくはリンク24、および、タグ削除器113もしくはもしくはツリーセレクタ116に転送する。

【0193】

タグ挿入器112は、リンク81より受信したフレームに対し、タグテーブル117の記述に従いタグを挿入し、フレーム転送器111に転送する。なお、タグテーブル117の記述によっては、タグを挿入せずに、受信したフレームをそのままフレーム転送器111に転送することもできるほか、同一フレームに0個以上の複数のタグを挿入し、もしくは、到着したフレームをコピーし、コピーしたそれぞれのフレームに対して、同一もしくは異なるタグを、0個以上の複数個挿入することも可能である。

【0194】

タグ削除器113は、フレーム転送器111より受信したフレームに付加され

ているタグを外し、リンク81に転送する。なお、設定によって、タグを外さずに、受信したフレームをそのままリンク81に転送することもできる。

#### 【0195】

フォワーディングテーブル114は、フレーム転送器111からの問い合わせに対し、MACアドレス、タグ、もしくは入力ポートの他、1つ以上の組み合わせをキーとして、1つ以上のフレーム転送先ポートをフレーム転送器111に返答する。キーおよび転送先ポートは、ツリーマネージャ1151もしくはツリーマネージャ1152により設定される。

#### 【0196】

分別器1150は、受信したフレームのタグに従い出力先ポートを決定し、前記フレームをツリーマネージャ1151、または、ツリーマネージャ1152に転送する。どのタグが付加されたフレームをツリーマネージャ1151もしくはツリーマネージャ1152に転送するかは、ツリーセレクタ116より設定することができる。

#### 【0197】

ツリーマネージャ1151は、ツリーセレクタ116の指示に従い、スパンニングツリーアルゴリズムを利用して、分別器1150よりBPDUsを受信し、またフレーム転送器111に対してBPDUsを送信し、フォワーディングテーブル114を設定する。さらに、ツリーセレクタ116から設定情報を受信し、BPDUsのパラメータとして利用する。また、BPDUsに含まれる制御情報を抽出し、ツリーセレクタ116に通知する。

#### 【0198】

ツリーマネージャ1152は、ツリーマネージャ1151と同様のツリーマネージャである。以後、ツリーマネージャ1151～ツリーマネージャ1152を代表してツリーマネージャ1151を用いて記述を行うが、ツリーマネージャ1151についての記述は、特に断りのない限りツリーマネージャ1152においても同様に適用可能である。

#### 【0199】

ツリーセレクタ116は、フレーム転送器111よりGVRP等の設定フレー

ムを受信し、ツリーマネージャ1151もしくは1152よりBPDUsに含まれる制御情報を受信し、リソースモニタ119よりリンク情報の通知を受け、または、設定インターフェース118からの設定通知を受け、設定フレームもしくは通知に含まれる情報に沿って、ツリーマネージャ1151、ツリーマネージャ1152、タグテーブル117の設定を行う。また、設定フレームをフレーム転送器111に送信する。

#### 【0200】

タグテーブル117は、タグ挿入器112からの問い合わせに対し、挿入すべきタグの情報、もしくは、タグを追加しないで転送する命令をタグ挿入器112に返答する。挿入するタグもしくは、タグ挿入なしで転送命令は、ツリーセレクタ116より設定される。同一フレームに0個以上の複数のタグを挿入する設定や、到着したフレームをコピーし、コピーしたそれぞれのフレームに対して、同一もしくは異なるタグを、0個以上の複数個挿入する設定も可能である。

#### 【0201】

設定インターフェース118は、シリアル接続もしくはTELNET等のコマンドラインインターフェース、もしくはWEBサーバ等を通じて、ユーザからのツリー選択命令や、ノード削除要求、リンクコスト、スパンニングツリーパラメータ値等を、ツリーセレクタ116に伝達する。

#### 【0202】

リソースモニタ119は、ノードの各リンクポートの状況を監視し、リンクの接続を検出した場合には、ツリーセレクタ116にリンクアップ通知を送信する。また、リンクを通過するフレームの累積バイト数、通過TCPセッション数、HTTPリクエスト数のうち1つ以上の値をカウントして保持し、ツリーセレクタ116の要求により、保持している値をツリーセレクタ116に通知するほか、ツリーセレクタ116からの命令により、保持している値を0にリセットする。さらに、ツリーセレクタ116よりあらかじめ指定された種類のフレーム通過を監視し、前記監視対象フレームが通過すると、ツリーセレクタ116に通知する。

#### 【0203】

図9は、本実施の形態の図8における、タグをキーとして出力ポートを決定す

る、フォワーディングテーブル114の構成例である。

【0204】

タグフィールド1141は、サーチのインデックスになるフィールドであり、このフィールドの情報と、受信したフレームのタグに書かれている内容が一致するかどうかを調べる。

【0205】

出力ポート1142は、受信したフレームのタグに書かれている内容とタグにフィールド1141の内容が一致した場合に、前記フレームをどのポートに転送すべきか記述するフィールドである。

【0206】

なお、本実施の形態は、本動作例に示すような、タグの内容によって転送先ポートを決定するタグフォワーディングを行う場合のみならず、従来より行われている、MACアドレスによって転送先を決定する通常のMACアドレス転送においても、同様に適用可能である。この場合は、出力ポートフィールド1142に、複数のポートが記載される。

【0207】

図10は、本発明の第1の実施の形態の図8における、ツリーマネージャ1151の構成を詳細に示した図である。ツリーマネージャ1151は、タグ削除器11511と、BPDUs送受信器11512と、タグ挿入器11513と、ツリーコントローラ11514と、ツリーテーブル11515含む。

【0208】

タグ削除器11511は、分別器1150より入力されたフレームに挿入されているタグを削除し、BPDUs送受信機11512に転送する。もし分別器1150より受信したフレームにタグが添付されていない場合には、受信したフレームをそのままBPDUs送受信機11512に転送する。

【0209】

BPDUs送受信器11512は、タグ削除器11511よりBPDUsを受信し、フレームに含まれる情報を、BPDUs受信通知によりツリーコントローラ11514に通知する。また、ツリーコントローラ11514よりBPDUs送信通知

を受け、フレームを生成してタグ挿入器11513に送信する。

【0210】

タグ挿入器11513は、BPDUs送受信機11512よりフレームを受信し、あらかじめ設定されているタグを挿入し、フレーム転送器111に送信する。なお、タグを挿入せずに、フレームをそのまま転送する設定も可能である。

【0211】

ツリーコントローラ11514は、以下に示す4つの機能を有する。

【0212】

(1) 停止動作(初期状態)：ツリーセレクタ116からの停止命令にしたがい、BPDUs送受信機11512へのBPDUs送信通知を停止する。また、ツリーテーブル11515に対して、すべてのリンクがdownしているとして、ポートの状態を登録する。

【0213】

(2) 開始動作：ツリーセレクタ116からの開始命令にしたがい、BPDUs送受信機11512へのBPDUs送信通知を開始する。また、開始命令に含まれる情報を元に、ツリーテーブル11515に対して、アップ状態のポートを登録する。

【0214】

(3) BPDUs受信動作：BPDUs送受信機11512からのBPDUs受信通知を受け取り、ツリーテーブル11515を更新する。また、BPDUs受信通知に含まれる現用系ツリーおよび予備系ツリーの識別情報を抽出し、ツリーセレクタ116内のメインコントローラ1164に通知する。

【0215】

(4) トポロジ更新動作：前記停止動作、前記開始動作、および前記BPDUs受信動作後に、従来技術1もしくは従来技術2に示すスパンニングツリープロトコルに従い、ツリーテーブル11515を参照し、必要であれば、ツリーテーブル11515の設定、フォワーディングテーブル114の設定および、BPDUsの送信を行う。送信するBPDUsには、現用系ツリーおよび予備系ツリーの識別情報も含まれる。また、ツリーセレクタ116に対し、ツリーの再計算の結果、ト

ポロジに変更が生じるか生じないかを通知する動作を行う。

【0216】

ツリーテーブル11515は、従来技術1もしくは従来技術2に示すスパニングツリープロトコルに必要な、ポートの状態およびノードの状態に関するパラメータが記述されているテーブルである。各ポートもしくはリンクの優先度並びにリンクコストも、このテーブルに記述されている。第1の実施の形態では、リンクコストとして、リンク帯域の太さを用いる場合を想定して説明する。

【0217】

図11は、本発明の第1の実施の形態の図8における、ツリーセレクタ116の構成を詳細に示した図である。ツリーセレクタ116は、タグ削除器1161と、GVRP送受信器1162と、タグ挿入器1163と、メインコントローラ1164と、安定タイマ1165と、到着間隔タイマ1166とを含む。

【0218】

タグ削除器1161は、フレーム転送器111より入力されたフレームに挿入されているタグを削除し、GVRP送受信機1162に転送する。もしフレーム転送器111より受信したフレームにタグが添付されていない場合には、受信したフレームをそのままGVRP送受信機1162に転送する。

【0219】

GVRP送受信器1162は、タグ削除器1161より制御フレームを受信し、フレームに含まれる情報を、GVRPフレーム受信通知によりメインコントローラ1164に通知する。また、メインコントローラ1164よりGVRPフレーム送信通知を受け、フレームを生成してタグ挿入器1163に送信する。

【0220】

タグ挿入器1163は、GVRP送受信機1162よりフレームを受信し、あらかじめ設定されているタグを挿入し、フレーム転送器111に送信する。なお、タグを挿入せずに、フレームをそのまま転送する設定も可能である。

【0221】

メインコントローラ1164は、以下に示す4つの機能を有する。

【0222】

(1) リンクアップ検出：メインコントローラ1164は、リソースモニタ119よりリンクアップ通知を受け、現在予備系となっているツリーマネージャに対して、リンクアップ（開始命令）を通知する。開始命令には、アップしているリンクの情報が格納される。また、開始命令送信後に、安定タイマ1165をセットする。安定タイマ1165の満了通知を受信すると、タグテーブル117に対して挿入タグの変更命令を通知し、さらに、予備系と現用系の登録を逆転させるよう、GVRP送受信機1162に対して、新ツリーのルートノードに対してツリー切り替え要求フレームを送信するよう命令する。さらに、安定タイマ1165をセットし、タイマ満了後に自ノードを旧ツリーに追加する。

#### 【0223】

(2) ノード削除要求受信：設定インターフェース118よりノード削除要求通知を受信すると、現在予備系となっているツリーマネージャに対して停止命令を送信する。また停止命令の送信後に、安定タイマ1165をセットし、安定タイマ1165の満了通知を受信すると、タグテーブル117に対して挿入タグの変更命令を通知し、さらに、予備系と現用系の登録を逆転させるよう、GVRP送受信機1162に対して、新ツリーのルートノードに対して利用タググループ変更GVRPフレームを送信するよう命令する。さらに、安定タイマ1165をセットし、タイマ満了後に設定インターフェース118に対して、自ノード削除許可の表示を行う。

#### 【0224】

(3) 利用タググループ変更GVRP受信：自ノードが新ツリーのルートノードである場合、利用タググループ変更GVRPフレームを受信すると、自ノードから送信するBPDUsに、現用系フラグを付加して送信するよう、ツリーマネージャ1151もしくはツリーマネージャ1152に対して命令する。さらに、GVRP送受信機1162に対して、旧ツリーのルートノードに対して、現用系フラグの添付を取り消すよう命令する。

#### 【0225】

(4) 現用系ビット変更通知受信：ツリーマネージャ1151もしくはツリーマネージャ1152は、BPDUsの受信時にBPDUsに付加された現用系フラグ

を確認し、自グループが現用系であるか予備系であるか確認し、メインコントローラに通知する。通知を受けた結果、もし、現用系と予備系に変更がある場合には、タグテーブル117に対して挿入タグ変更通知を送信して、現用系と予備系の登録を逆転させる。

#### 【0226】

(5) 指定フレームの通過通知受信：リソースモニタ119より、あらかじめ設定した監視対象フレームが通過したことの通知を受信すると、到着間隔タイマ1166に対してセット通知を送信する。到着間隔タイマ1166から、タイマ満了通知が到着した場合は、監視対象フレームの到着間隔がセット通知によって設定した時間より長いことがわかる。これによりBPDUの到着間隔が長くなつたことや、予備系を流れるフレームの到着間隔が長くなつたことなどを検出できる。

#### 【0227】

安定タイマ1165は、メインコントローラ1164より送信されたセット通知の受信から、あらかじめ設定された時間が経過すると、メインコントローラ1164に対してタイマ満了通知を送信する。

#### 【0228】

到着間隔タイマ1166は、メインコントローラ1164からセット命令を受信すると、現在保持している時間を0にリセットしてタイマを作動させ、セット命令によって指定された時間が経過すると、メインコントローラ1164に対してタイマ満了通知を送信する。

#### 【0229】

図12は、本発明の第1の実施の形態の図11における、メインコントローラ1164の状態遷移を詳細に示した流れ図である。

#### 【0230】

以降、2系統存在するスパニングツリーのうち、新しくネットワークに侵入するデータフレームの転送に利用しているスパニングツリーを現用ツリーもしくは現用系ツリー、現用ツリーとなっていないツリーを予備用ツリーもしくは予備系ツリーと表現する。

## 【0231】

また、現用系ツリーの生成を行うツリーマネージャを現用系ツリーマネージャ、予備系ツリーの生成を行うツリーマネージャを、予備系ツリーマネージャと呼ぶ。

## 【0232】

状態11641は、現用系ツリーマネージャが、ツリーマネージャ1151であるか、ツリーマネージャ1152であるか判別できていない状態であり、かつ、ツリーマネージャ1151内のBPDUs送受信機11512および、ツリーマネージャ1152内のBPDUs送受信機11522のBPDUs送信機能は無効にされ、BPDUs受信機能のみが有効になっている状態である。

## 【0233】

状態11642は、現用系ツリーマネージャが、ツリーマネージャ1151であり、予備系ツリーマネージャがツリーマネージャ1152であり、ツリーマネージャ1151内のBPDUs送受信機11512のBPDUs送信機能が無効にされ、ツリーマネージャ1152内のBPDUs送受信機11522のBPDUs送信機能も無効にされている状態である。なお、BPDUs送受信機11512およびBPDUs送受信機11522のBPDUs受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

## 【0234】

状態11643は、現用系ツリーマネージャが、ツリーマネージャ1151であり、予備系ツリーマネージャがツリーマネージャ1152であり、ツリーマネージャ1151内のBPDUs送受信機11512のBPDUs送信機能が無効にされ、ツリーマネージャ1152内のBPDUs送受信機11522のBPDUs送信機能は有効にされている状態である。なお、BPDUs送受信機11512およびBPDUs送受信機11522のBPDUs受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

## 【0235】

状態11644は、現用系ツリーマネージャが、ツリーマネージャ1152であり、予備系ツリーマネージャがツリーマネージャ1151であり、ツリーマネ

ージャ1152内のBPDU送受信機11522のBPDU送信機能が有効にされ、ツリーマネージャ1151内のBPDU送受信機11512のBPDU送信機能は無効にされている状態である。なお、BPDU送受信機11512およびBPDU送受信機11522のBPDU受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

#### 【0236】

状態11645は、現用系ツリーマネージャが、ツリーマネージャ1152であり、予備系ツリーマネージャがツリーマネージャ1151であり、ツリーマネージャ1152内のBPDU送受信機11522のBPDU送信機能が有効にされ、ツリーマネージャ1151内のBPDU送受信機11512のBPDU送信機能も有効にされている状態である。なお、BPDU送受信機11512およびBPDU送受信機11522のBPDU受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

#### 【0237】

状態11646は、現用系ツリーマネージャが、ツリーマネージャ1151であり、予備系ツリーマネージャがツリーマネージャ1152であり、ツリーマネージャ1151内のBPDU送受信機11512のBPDU送信機能が有効にされ、ツリーマネージャ1152内のBPDU送受信機11522のBPDU送信機能も有効にされている状態である。なお、BPDU送受信機11512およびBPDU送受信機11522のBPDU受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

#### 【0238】

状態11647は、現用系ツリーマネージャが、ツリーマネージャ1151であり、予備系ツリーマネージャがツリーマネージャ1152であり、ツリーマネージャ1151内のBPDU送受信機11512のBPDU送信機能が有効にされ、ツリーマネージャ1152内のBPDU送受信機11522のBPDU送信機能が無効にされている状態である。なお、BPDU送受信機11512およびBPDU送受信機11522のBPDU受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

## 【0239】

状態11648は、現用系ツリーマネージャが、ツリーマネージャ1152であり、予備系ツリーマネージャがツリーマネージャ1151であり、ツリーマネージャ1152内のBPDU送受信機11522のBPDU送信機能が無効にされ、ツリーマネージャ1151内のBPDU送受信機11512のBPDU送信機能が有効にされている状態である。なお、BPDU送受信機11512およびBPDU送受信機11522のBPDU受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

## 【0240】

状態11649は、現用系ツリーマネージャが、ツリーマネージャ1152であり、予備系ツリーマネージャがツリーマネージャ1151であり、ツリーマネージャ1152内のBPDU送受信機11522のBPDU送信機能が無効にされ、ツリーマネージャ1151内のBPDU送受信機11512のBPDU送信機能が無効にされている状態である。なお、BPDU送受信機11512およびBPDU送受信機11522のBPDU受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

## 【0241】

次に、図12を参照して、メインコントローラ1164の動作について説明する。

## 【0242】

メインコントローラ1164は、リソースモニタ119より新たにネットワークに接続されたことの通知を受信すると、ツリーマネージャ1151もしくはツリーマネージャ1152からの現用系通知の到着を待つ。ツリーマネージャ1151もしくはツリーマネージャ1152よりBPDUに含まれる現用系通知を受信すると、前記通知において指定されたツリーマネージャ1151もしくはツリーマネージャ1152を現用、予備用に指定されたツリーマネージャ1151もしくは1152を予備用に設定し、状態11642もしくは状態11649に遷移する。ここでは、状態11642に遷移した場合を例に解説するが、以後の解説は状態11649に遷移した場合においても同様である。（状態11641）

## 【0243】

メインコントローラ1164は、ツリーマネージャ1151を現用、ツリーマネージャ1152を予備用に設定する。さらに、前記各ツリーマネージャ1151および1152に対して、BPDU送信停止命令を出す。（状態11642）

## 【0244】

メインコントローラ1164は、状態11642において、もし設定インタフェース118よりノードの追加要求を受信した場合は、状態11643に遷移する。また、もしツリーマネージャ1151もしくはツリーマネージャ1152より現用系通知を受信し、現用系と予備系の関係に変更があった場合には、状態11649に遷移する。（状態11642）

## 【0245】

メインコントローラ1164は、ツリーマネージャ1152に対してリンクupの通知を行い、同時に、ツリーマネージャ1152に対してBPDUの送信許可を行う。さらに、安定タイマ1165を作動させる。（状態11643）

## 【0246】

メインコントローラ1164は、安定タイマ1165よりタイマ満了通知を受信すると、現用に登録されているツリーマネージャ1151と、予備用に登録されているツリーマネージャ1152を入れ替え、新しく現用をツリーマネージャ1152、予備用をツリーマネージャ1151とする。さらに、GVRP送受信機1162を通じて、新ツリーのルートノードに対して、利用タググループ変更通知を送信する。この利用タググループ変更通知の内容は、BPDUに反映され、全ノードに伝達される。その後、安定タイマ1165を作動させる。（状態11644）

## 【0247】

メインコントローラ1164は、安定タイマ1165よりタイマ満了通知を受信すると、ツリーマネージャ1151に対してリンクupの通知を行い、同時に、ツリーマネージャ1152に対してBPDUの送信許可を行う。通常は、この状態で安定する。（状態11645）

## 【0248】

メインコントローラ1164は、状態11645において、ツリーマネージャ1151もしくはツリーマネージャ1152よりBPDUsに含まれる現用系通知を受信し、もし現用系と予備系の関係に変更が生じた場合には、状態11646に遷移する。そして、現用に登録されているツリーマネージャ1152と、予備用に登録されているツリーマネージャ1151を入れ替え、新しく現用をツリーマネージャ1151、予備用をツリーマネージャ1152とする。（状態11645）

#### 【0249】

メインコントローラ1164は、状態11645において、もし設定インタフェース118より、ノード削除要求を受信した場合は、状態11644に遷移する。（状態11645）

#### 【0250】

メインコントローラ1164は、ツリーマネージャ1151に対して、接続されている全リンクリンクdownの通知を行い、同時に、ツリーマネージャ1151に対してBPDUsの送信停止命令を出す。さらに、安定タイマ1165を作動させる。（状態11644）

#### 【0251】

メインコントローラ1164は、安定タイマ1165よりタイマ満了通知を受信すると、現用に登録されているツリーマネージャ1152と、予備用に登録されているツリーマネージャ1151を入れ替え、新しく現用をツリーマネージャ1151、予備用をツリーマネージャ1152とする。さらに、GVRP送受信機1162を通じて、新ツリーのルートノードに対して、利用タググループ変更通知を送信する。この、利用タググループ変更通知の内容は、BPDUsに反映され、全ノードに伝達される。その後、安定タイマ1165を作動させる。（状態11643）

#### 【0252】

メインコントローラ1164は、安定タイマ1165よりタイマ満了通知を受信すると、ツリーマネージャ1151に対してリンクdownの通知を行い、同時に、ツリーマネージャ1151に対してBPDUsの受信停止命令を出す。さら

に、無条件で状態11641に遷移し、ノード切離しまで待機する。（状態11642）

図13は、本実施の形態の図8における、宛先MACアドレスをキーとして挿入するタグを決定する、タグテーブル117の構成例である。

#### 【0253】

宛先MACアドレス1171は、サーチのインデックスになるフィールドであり、このフィールドの情報と、受信したフレームの宛先MACアドレスフィールド、すなわち、MAC DAフィールドに書かれている内容が一致するかどうか調べ、一致した場合には、前記受信フレームに対して、挿入タグフィールド1172に記載のタグを挿入する。

#### 【0254】

挿入タグフィールド1172は、宛先MACアドレスフィールド1171に対する、挿入すべきタグが記載されるフィールドである。本実施の形態では、現在において現用系となっているタググループのタグが挿入される。この挿入タグフィールド1172は、ツリーセレクタ116によって、現在現用系となっているタグに書き換えられる。

#### 【0255】

図8、図14、図15及び図16を参照して、本実施の形態において、ノード17を追加する場合の動作について、具体例を用いて詳細に説明する。

#### 【0256】

初期状態（ノード17がリンク29、30に接続される前の状態）における、2系統のスパニングツリーは、同じ接続関係となる。これは、スパニングツリーを設定するプロトコルは、ノード、リンクのプライオリティ値等の情報に基づいてスパニングツリーを設定するため、同一のネットワークについてスパニングツリーを2つ設定すると結果として同じ接続関係となるものである。この初期状態で、何れか一方のスパニングツリーを現用、他方を予備として設定し（具体的には、ツリーマネージャ1151、1152がツリーテーブル11515を書き換え、ツリーセレクタ116がタグテーブル117を書き換えることで各設定を行う）、このネットワークは現用のスパニングツリーを用いて運用される。

## 【0257】

図10に示すネットワークにおいて、初期状態で、太線で示すスパニングツリー51が2つ設定されている場合を想定する。

## 【0258】

図14を参照すると、本動作例では、ノード11～17、リンク21～30、およびツリー51を有する。ただし、ノード17、リンク29およびリンク30は、初期状態では接続されていない。

## 【0259】

また、図15は、本動作例において、ノード17が追加された後のスパニングツリー52の状態を示す。ツリー52は図11において太線で示されている。

## 【0260】

ノード11～16の全ノード、全ポートが所属するタググループが2つすでに設定されており、1つめのタググループをタググループ41、2つめのタググループをタググループ42と呼ぶ。

## 【0261】

なお、基本的には全ノード、全ポートを2つのタググループに加入させるが、一部のポートもしくはノードのみで構成させるタググループを作成しても良い。以降、全ノード、全ポートが2つのタググループに加入するものとして説明する。

## 【0262】

ノード11～16には、独立して動作するスパニングツリー回路が2つ存在し、タググループ41上で動作するスパニングツリーをツリー51、タググループ42上で動作するスパニングツリーをツリー52と呼ぶ。

## 【0263】

スパニングツリーは必ず2系統作成するが、タググループは必ずしも2つ作成する必要は無い。前記タググループ41のみ設定してタググループ42を利用せず、タググループ41上で動作するスパニングツリーをツリー51、タググループに属さないで動作するスパニングツリーをツリー52と呼ぶこともできる。また逆に、前記タググループ42のみ設定してタググループ41を利用せず、タグ

グループに属さないで動作するスパニングツリーをツリー51、タググループ42上で動作するスパニングツリーをツリー52と呼ぶこともできる。

#### 【0264】

ここでは、特にタググループ41とタググループ42の両方を用いた場合について説明を行うが、タググループ41及びタググループ42の両方を用いた場合の動作は、タググループ41のみ、もしくはタググループ42のみを用いた場合においても、同様に適用可能である。

#### 【0265】

このネットワークでは、設定インターフェース118からの初期設定によって、ツリー51が現用系のスパニングツリーとなり、ツリー52が予備系のスパニングツリーとなるように設定されており、ツリー51のBPDUsには現用系フラグおよびタググループ41のタグが付き、ツリー52のBPDUsには予備系フラグおよびタググループ42のタグが付く。

#### 【0266】

全ノードが、現用系フラグ又は予備系フラグが付されたBPDUsフレームをIEEE Std 802.1DやIEEE Std 802.1wで規定される一定の周期で互いに送信し合うことにより、ツリー51である現用系のスパニングツリー又はツリー52である予備系のスパニングツリーが構築される。

#### 【0267】

上記現用系フラグ又は予備系フラグは、図5に示したBPDUsフレームのフィールドのうち、例えばタグ領域2203、Type2204、BPDUType22053等のフィールドを使用することによって表すことができる。

#### 【0268】

現在、ネットワークの開始から十分に時間が経過し、ツリー51およびツリー52の各ツリーは、タググループ41およびタググループ42のタグが付加されたBPDUsフレームの交換が十分に行われた結果、共にノード11をルートノードとして安定しているとする。

#### 【0269】

安定とは、スパニングツリーのツリー構造が、十分に長い時間変化しない状態

になっている状態のことを言う。

【0270】

ツリー51のBPDUsにはタググループ41のタグが、ツリー52のBPDUsにはタググループ42のタグが付く。具体的には、ツリー51を用いて伝送されるBPDUsのタグ領域2203には、このBPDUsがタググループ41に属することを示す値が記載され、ツリー52を用いて伝送されるBPDUsのタグ領域2203には、このBPDUsがタググループ42に属することを示す値が記載される。

【0271】

また、現時点では、ツリー51が現用系として設定されているので、クライアントからノード11～16に送信されたデータには、タグ挿入器112によって、タググループ41のタグが付加されている。具体的には、データ信号のタグ領域に、タググループ41に属することを示す値が記載される。そして、このタグが付加されたデータは、フレーム転送器111によって、現時点で、現用系として設定されているツリー51に沿って転送されている。

【0272】

ノード17は、リンク29およびリンク30と接続されると、どのタググループにも加入せず、BPDUsの受信を開始する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11641となる。）

【0273】

ノード17は、各タググループのBPDUsを受信すると、現時点での現用系がタググループ41で、現時点での予備系がタググループ42であると確認する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11642である。）そして、自ノードをタググループ42にのみ参加させてBPDUsの送受信を行い、タググループ41ではBPDUsは受信のみを行い、送信を行わないように設定する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11643である。）

【0274】

ノード17が追加されると、タググループ42のメンバに変更が発生するため

、スパンディングツリープロトコルにより、ツリー52を更新する動作が開始される。すなわち、ノード17がBPDUsを送信して、それを隣接ノードが受信すると、隣接ノードはトポロジ状況に変更があったと認識し、ツリー52の更新動作を開始する。タググループ41のメンバには変更が無いため、ツリー51は更新されない。

## 【0275】

クライアントから送信されるフレームには、各ノード11～16によって、今までどおりタググループ41のタグが付加され、ツリー51に沿って転送され続ける。

## 【0276】

ここで、段落0135に記載したツリー52の更新動作によって、ツリー52がルートノードをノード11として、安定したとする。このときのツリー52の構成を図15に示す。

## 【0277】

ノード17は、ネットワークに接続されてから一定時間経過後に、ツリー52が安定したと判断し、利用タググループ変更通知を、ツリー52のルートノードである、ノード11に送信し、ツリー52を予備系から現用系に遷移させるよう命令する。この命令には、例えば制御フレーム(GVRP)を利用する。(このときのノード17のメインコントローラ1164の状態は、図12の状態11644である。)

## 【0278】

なお、ツリー52が安定したことの検出は、ノード17がネットワークに接続されてから一定時間経過することで検出するほか、ノード17におけるツリー52のBPDUs到着間隔が一定時間以上となることで検出することもできる。

## 【0279】

利用タググループ変更通知を受信したノード11は、ツリー51のルートノードであるノード11に対して利用タググループ変更通知を送信し、ツリー51を予備系に遷移させるよう命令する。この利用タググループ変更通知には、例えば制御フレーム(GVRP)を利用する。そして、タググループ42のタグを付け

、ツリー52のために送信するBPDUsに、現用系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

#### 【0280】

ツリー51のノード11は、利用タググループ変更通知を受信し、ツリー51を予備系に遷移させ、タググループ41のフラグを付け、ツリー51のために送信するBPDUsに、予備系フラグを立てる。この予備系フラグは、例えば、図1のBPDUsのうちの予め決めてあるフィールドに設定される。設定は、ツリーマネージャ1151、1152がツリーテーブル11515を書き換え、ツリーセレクタ116がタグテーブル117を書き換えることで実行される。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

#### 【0281】

ノード11～17は、タググループ42のタグが付いたBPDUsに現用系フラグが付加させたのを確認し、クライアントより転送されてくるフレームに付加するタグを、タググループ41からタググループ42に切り替える。このとき、タグテーブル117内の、挿入タグフィールド1172を書き換えられる。このタグが付加されたフレームは、ツリー52に沿って転送される。

#### 【0282】

以上の切り替えが完了してしばらく経つと、ツリー51を流れるフレームはなくなる。

#### 【0283】

ノード17は、利用タググループ変更通知の送信から一定時間経過後に、タググループ41のタグを付加するノードが存在しなくなったと判断し、自ノードをタググループ41に加入させ、次のトポロジ変更に備える。自ノードをタググループ41に加入させるために、ノード17は、ツリーマネージャ1151内のツリーコントローラに対して、BPDUs送信を許可し、BPDUs送受信機11512からのBPDUs送信を許可する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11645となる。）

このとき、スパニングツリー51について、再構成が行われるので、スパニングツリー51に着目してネットワークを見ると、ネットワークは停止する。しか

し、この間もネットワークにおける通信は、スパンニングツリー52を用いて行われているので、ノード17の追加に伴って、輻輳、フレームの到着遅延等の問題は発生しない。

#### 【0284】

なお、ノード17をタググループ41に加入させる動作は、ツリー52のルートノードであるノード11、もしくは、ツリー51のルートノードであるノード11が行っても良い。

#### 【0285】

ツリー52のルートノードであるノード11が、ノード17をタググループ41に加入させる場合は、ノード11がノード17から利用タググループ変更通知を受信してから一定時間経過後に、タググループ41のタグを付加するノードが存在しなくなったと判断し、ノード17に対してGVRPフレームを送信し、タググループ41に加入するよう命令する。

#### 【0286】

ツリー51のルートノードであるノード11が、ノード17をタググループ41に加入させる場合は、ノード11がノード11から利用タググループ変更通知を受信してから一定時間経過後に、タググループ41のタグを付加するノードが存在しなくなったと判断し、ノード17に対してGVRPフレームを送信し、タググループ41に加入するよう命令する。この場合は、ツリー51のノード11は、直接（ツリー52のノード11を介さずに）、ノード17に対してGVRPフレームを送信する。ノード17をタググループ41に加入させる動作では、BPDUに挿入するフラグに変更が生じないので、ルートノードを介す必要はないからである。

#### 【0287】

以上のようにして、ネットワークを停止させることなく、ノード17を追加することができた。以降ノードの追加の際は、同様の動作を繰り返す。ただし、上記説明文中のタググループ41とタググループ42は適宜入れ替わる。

#### 【0288】

図16は、以上に述べたノード17の追加動作を表すシーケンス図である。

【0289】

矢印31は、タググループ41を示すタグが挿入された、現用系フラグ付きB PDUの流れを示す。

【0290】

矢印32は、タググループ42を示すタグが挿入された、予備系フラグ付きB PDUの流れを示す。

【0291】

矢印33は、タググループ42を示すタグが挿入された、現用系フラグ付きB PDUの流れを示す。

【0292】

矢印34は、タググループ41を示すタグが挿入された、予備系フラグ付きB PDUの流れを示す。

【0293】

矢印35は、タググループを示すタグが挿入されていない、GVRPフレーム等による利用タググループ変更通知の流れを示す。

【0294】

次に、図15および図14を参照して、本実施の形態において、ノード17を削除する場合の動作について、具体例を用いて詳細に説明する。

【0295】

図15および図14を参照すると、本動作例では、ノード11～17、およびリンク21～30を有する。

【0296】

ノード11～17の全ポートが所属するタググループが2つすでに設定されており、1つめのタググループをタググループ41、2つめのタググループをタググループ42と呼ぶ。

【0297】

ノード11～17には、独立して動作するスパンニングツリー経路が2つ存在し、タググループ41上で動作するスパンニングツリーをツリー51、タググループ42上で動作するスパンニングツリーをツリー52と呼ぶ。

## 【0298】

図15では、ツリー52は太線で表され、ノード11をルートノードとして安定しているとする。

## 【0299】

ツリー51のBPDUsにはタググループ41のタグが、ツリー52のBPDUsにはタググループ42のタグがつく。

## 【0300】

現在、クライアントからノード11～17に送信されたデータには、ツリー52が現用系であるので、タググループ42のタグが付加されている。このタグが付加されたデータは、ツリー52に沿って転送されている。

## 【0301】

ノード17は、各タググループのBPDUsを受信しており、現時点での現用系がタググループ42で、現時点での予備系がタググループ41であるとすでに確認しているとする。

## 【0302】

ノード17は、設定インターフェースもしくはその他の手段により削除要求を受けると、自ノード17を現用系であるタググループ42にのみ参加させ、タググループ41には参加させないように設定する。このとき、ノード17は、タググループ41のBPDUs送信を停止する。

## 【0303】

この設定により、ノード17の隣接ノードにおいてBPDUsを受信しなくなることでノード17が削除されたことが認識され、タググループ41のメンバに変更が発生するため、ツリー51を更新する動作が開始される。タググループ42のメンバには変更が無いため、ツリー52は更新されない。

## 【0304】

クライアントからのフレームは、ノード11～ノード17で、今までどおりタググループ42のタグが付加され、ツリー52に沿って転送され続ける。

## 【0305】

ここで、ツリー51はルートノードをノード11として、ノード17を参加さ

せないで安定した状態を示している。ツリー51の構成を図14に示す。

【0306】

なお、ここでいう安定とは、スパニングツリーのツリー構造が、十分に長い時間変化しない状態になっている状態のことを言う。

【0307】

ノード17は、タググループ41不参加の設定から一定時間経過後に、ツリー51が安定したと判断し、利用タググループ変更通知を、ツリー51のルートノードである、ノード11に送信し、ツリー51を予備系から現用系に遷移させるよう命令する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11644である。）

【0308】

利用タググループ変更通知を受信したノード11は、ツリー52のルートノードであるノード11に対して利用タググループ変更通知を送信し、ツリー52を予備系に遷移させるよう命令する。そして、タググループ41のフラグを付け、ツリー51のために送信するBPDUsに、現用系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

【0309】

ツリー51のノード11は、利用タググループ変更通知を受信し、ツリー52を予備系に遷移させ、タググループ42のフラグを付け、ツリー52のために送信するBPDUsに、予備系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

【0310】

ノード11～17は、タググループ41のタグが付いたBPDUsに現用系フラグが付加させたのを確認し、クライアントより転送されてくるフレームに付加するタグを、タググループ42からタググループ41に切り替える。このタグが付加されたフレームは、ツリー51に沿って転送される。

【0311】

以上の切り替えが完了してしばらく経つと、ツリー52を流れるフレームはなくなる。

## 【0312】

ノード17は、利用タググループ変更通知の送信から一定時間経過後に、タググループ42のタグを付加するノードが存在しなくなったと判断し、自ノードをネットワークから削除しても良いという、削除許可の通知を、設定インターフェース118に出力する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11643である。）

## 【0313】

以降ノードの削除の際は、同様の動作を繰り返す。ただし、上記説明文中のタググループ41とタググループ42は適宜入れ替わる形となる。

## 【0314】

図14および図15を参照して、本実施の形態において、タグをBPDUsのみに付加し、データにはタグを付加しない場合における、ノード17を追加する場合の動作について、具体例を用いて詳細に説明する。

## 【0315】

図14では、ツリー51は太線で表され、ノード11をルートノードとして安定しているとする。安定とは、スパニングツリーのツリー構造が、十分に長い時間変化しない状態になっている状態のことを言う。

## 【0316】

ツリー51のBPDUsにはタググループ41のタグが、ツリー52のBPDUsにはタググループ42のタグがつく。

## 【0317】

現在、クライアントからノード11～16に送信されたデータには、何もタグは付加されない。データは、各ノードのフォワーディングテーブルにおける設定にしたがって、ツリー51に沿って転送されている。

## 【0318】

ノード17は、リンク29およびリンク30と接続されると、どのタググループにも加入せず、BPDUsの受信を開始する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11641である。）

## 【0319】

ノード17は、各タググループのBPDUsを受信すると、BPDUsに付加されたタグ内のフラグにより、現時点での現用系がタググループ41で、現時点での予備系がタググループ42であると確認する。そして、タググループ42でのみBPDUsの送受信を行い、タググループ41ではBPDUsは受信のみを行い、送信を行わないように設定する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11642である。）

#### 【0320】

ノード17が追加されると、タググループ42のメンバに変更が発生するため、スパンニングツリープロトコルにより、ツリー52を更新する動作が開始される。タググループ41のメンバには変更が無いため、ツリー51は更新されない。（このときのノード17のメインコントローラ1164の状態は、図12の状態11643である。）

#### 【0321】

クライアントから送信されるフレームには、今までどおりタグは付加されず、ツリー51に沿って転送され続ける。

#### 【0322】

ここで、ツリー52はルートノードをノード11として、安定したとする。ツリー52の構成を図15に示す。

#### 【0323】

ノード17は、ネットワークに接続されてから一定時間経過後に、ツリー52が安定したと判断し、利用タググループ変更通知を、ツリー52のルートノードである、ノード11に送信し、ツリー52を予備系から現用系に遷移させるよう命令する。（このときのノード17のメインコントローラ1164の状態は、図12の状態11644である。）

#### 【0324】

利用タググループ変更通知を受信したツリー52のノード11は、ツリー51のルートノードであるノード11に対して利用タググループ変更通知を送信し、ツリー51を予備系に遷移させるよう命令する。そして、タググループ42のフラグを付け、ツリー52のために送信するBPDUsに、現用系フラグを立てる。

B P D Uは、各ノードで転送されながら、全ノードに伝播する。

【0325】

ツリー51のノード11は、利用タググループ変更通知を受信し、ツリー51を予備系に遷移させ、タググループ41のフラグを付け、ツリー51のために送信するB P D Uに、予備系フラグを立てる。B P D Uは、各ノードで転送されながら、全ノードに伝播する。

【0326】

ノード11～17は、タググループ42のタグが付いたB P D Uに現用系フラグが付加させたのを確認し、ルーティングテーブルを、ツリー52に従った設定に変更する。これにより、前記ノードにおいて転送されるフレームは、ツリー52に沿って転送されるようになる。

【0327】

すべてのノードのルーティングテーブルが、ツリー52利用のものに切り替わると、ツリー51を流れるフレームはなくなる。

【0328】

ノード17は、利用タググループ変更通知の送信から一定時間経過後に、タググループ41に従ったテーブル設定を行うノードが存在しなくなったと判断し、自ノードをタググループ41に加入させ、次のトポロジ変更に備える。（このときのノード17のメインコントローラ1164の状態は、図12の状態11644である。）

【0329】

以上のようにして、ネットワークを停止させることなく、ノード17を追加することができた。以降ノードの追加の際は、同様の動作を繰り返す。ただし、上記説明文中のタググループ41とタググループ42は適宜入れ替わる。

【0330】

以上の説明では、タイマーを用いて、ツリーの安定状態を確認していたが、安定状態の確認方法はこれに限らず、以下に示すように、B P D Uまたはフレームの到着間隔を計測し、それによって安定状態を確認することも可能である。

【0331】

図14では、ツリー51は太線で表され、ノード11をルートノードとして安定しているとする。

【0332】

ツリー51のBPDUsにはタググループ41のタグが、ツリー52のBPDUsにはタググループ42のタグがつく。

【0333】

現在、クライアントからノード11～16に送信されたデータには、タググループ41のタグが付加されている。このタグが付加されたデータは、ツリー51に沿って転送されている。

【0334】

ノード17は、リンク29およびリンク30と接続されると、どのタググループにも加入せず、BPDUsの受信を開始する。

【0335】

ノード17は、各タググループのBPDUsを受信すると、現時点での現用系がタググループ41で、現時点での予備系がタググループ42であると確認する。そして、自ノードをタググループ42にのみ加入させてBPDUsの送受信を行い、タググループ41ではBPDUsは受信のみを行い、送信を行わないように設定する。そして、タググループ42のルートノードであるノード11および、タググループ41のルートノードであるノード11のそれぞれに対して、ノード17が追加されたことを通知するフレームを送信する。

【0336】

ノード17が追加されると、タググループ42のメンバに変更が発生するため、スパンニングツリープロトコルにより、ツリー52を更新する動作が開始される。タググループ41のメンバには変更が無いため、ツリー51は更新されない。

【0337】

クライアントから送信されるフレームには、今までどおりタググループ41のタグが付加され、ツリー51に沿って転送され続ける。

【0338】

ここで、ツリー52はルートノードをノード11として、安定したとする。ツ

ツリー52の構成を図15に示す。

#### 【0339】

ツリー52のルートノードであるノード11は、ツリー52のBPDUs到着間隔が一定時間以上となることを検出すると、ツリー52が安定したと判断し、ツリー51のルートノードであるノード11に対して利用タググループ変更通知を送信し、ツリー51を予備系に遷移させるよう命令する。そして、タググループ42のフラグを付け、ツリー52のために送信するBPDUsに、現用系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

#### 【0340】

なお、ツリー52の安定検出は、ツリー51のルートノードであるノード11が行っても良い。この場合、ツリー51のルートノードであるノード11は、ツリー52のBPDUs到着間隔が一定時間以上となることを検出すると、ツリー52が安定したと判断し、ツリー52のルートノードであるノード11に対して利用タググループ変更通知を送信し、ツリー52を予備系に遷移させるよう命令する。ツリー51のルートノードであるノード11は、タググループ42のフラグを付け、ツリー52のために送信するBPDUsに、現用系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

#### 【0341】

ノード11は、利用タググループ変更通知を受信し、ツリー51を予備系に遷移させ、タググループ41のフラグを付け、ツリー51のために送信するBPDUsに、予備系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

#### 【0342】

ノード11～17は、タググループ42のタグが付いたBPDUsに現用系フラグが付加させたのを確認し、クライアントより転送されてくるフレームに付加するタグを、タググループ41からタググループ42に切り替える。このタグが付加されたフレームは、ツリー52に沿って転送される。

#### 【0343】

以上の切り替えが完了してしばらく経つと、ツリー51を流れるフレームはな

くなる。

【0344】

タググループ41のルートノードであるノード11は、タググループ41のタグを付加してツリー51を流れるフレームの到着間隔が一定時間以上になると、タググループ41のタグを付加するノードが存在しなくなったと判断し、ノード17に対してGVRPフレームを送信し、タググループ41に加入するよう命令し、次のトポロジ変更に備える。

【0345】

なお、ノード17をタググループ41に加入させる動作は、ツリー52のルートノードであるノード11が行っても良い。

【0346】

ツリー52のルートノードであるノード11が、ノード17をタググループ41に加入させる場合は、タググループ41のタグを付加してツリー51を流れるフレームの到着間隔が一定時間以上になると、タググループ41のタグを付加するノードが存在しなくなったと判断し、ノード17に対してGVRPフレームを送信し、タググループ41に加入するよう命令し、次のトポロジ変更に備える。

【0347】

以上のようにして、ネットワークを停止させることなく、ノード17を追加することができた。以降ノードの追加の際は、同様の動作を繰り返す。ただし、上記説明文中のタググループ41とタググループ42は適宜入れ替わる。

【0348】

図14および図15を参照して、本実施の形態において、予備系への移行完了通知を受信することによって、予備系に遷移したことを検知する場合における、ノード17を追加する場合の動作について、具体例を用いて詳細に説明する。

【0349】

図14では、ツリー51は太線で表され、ノード11をルートノードとして指定しているとする。

【0350】

ツリー51のBPDUsにはタググループ41のタグが、ツリー52のBPDUs

にはタググループ4 2のタグがつく。

【0351】

現在、クライアントからノード11～16に送信されたデータには、タググループ4 1のタグが付加されている。このタグが付加されたデータは、ツリー5 1に沿って転送されている。

【0352】

ノード17は、リンク29およびリンク30と接続されると、どのタググループにも加入せず、BPDUsの受信を開始する。

【0353】

ノード17は、各タググループのBPDUsを受信すると、現時点での現用系がタググループ4 1で、現時点での予備系がタググループ4 2であると確認する。そして、自ノードをタググループ4 2にのみBPDUsの送受信を行い、タググループ4 1ではBPDUsは受信のみを行い、送信を行わないように設定する。

【0354】

ノード17が追加されると、タググループ4 2のメンバに変更が発生するため、スパンニングツリープロトコルにより、ツリー5 2を更新する動作が開始される。タググループ4 1のメンバには変更が無いため、ツリー5 1は更新されない。

【0355】

クライアントから送信されるフレームには、今までどおりタググループ4 1のタグが付加され、ツリー5 1に沿って転送され続ける。

【0356】

ここで、ツリー5 2はルートノードをノード11として、安定したとする。ツリー5 2の構成を図15に示す。

【0357】

ノード17は、ネットワークに接続されてから一定時間経過後に、ツリー5 2が安定したと判断し、利用タググループ変更通知を、ツリー5 2のルートノードである、ノード11に送信し、ツリー5 2を予備系から現用系に遷移させるよう命令する。

【0358】

利用タググループ変更通知を受信したノード11は、ツリー51のルートノードであるノード11に対して利用タググループ変更通知を送信し、ツリー51を予備系に遷移させるよう命令する。そして、タググループ42のフラグを付け、ツリー52のために送信するBPDUsに、現用系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

#### 【0359】

ノード11は、利用タググループ変更通知を受信し、ツリー51を予備系に遷移させ、タググループ41のフラグを付け、ツリー51のために送信するBPDUsに、予備系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

#### 【0360】

ノード11～17は、タググループ42のタグが付いたBPDUsに現用系フラグが付加させたのを確認し、クライアントより転送されてくるフレームに付加するタグを、タググループ41からタググループ42に切り替え、さらに切替完了通知をツリー52のルートノードであるノード11に対して送信する。

#### 【0361】

以上の切り替えが完了してしばらく経つと、ツリー51を流れるフレームはなくなる。

#### 【0362】

ノード11は、ノード11～17のすべてのノードから切替完了通知を受信すると、タググループ41のタグを付加するノードが存在しなくなったと判断し、ノード17に対してGVRPフレームを送信し、タググループ41に加入するよう命令する。

#### 【0363】

なお、ノード17をタググループ41に加入させる動作は、新規に追加されたノードであるノード17、もしくは、ツリー51のルートノードであるノード11が行っても良い。

#### 【0364】

新規に追加されたノードであるノード17が、ノード17自身をタググループ

4.1に加入させる場合は、ノード11～16が、タググループ4.2のタグが付いたBPDUsに現用系フラグが付加させたのを確認し、クライアントより転送されてくるフレームに付加するタグを、タググループ4.1からタググループ4.2に切り替える際に、切替完了通知を新規に追加されたノードであるノード17に対して送信し、ノード17が、ノード11～16のすべてのノードから切替完了通知を受信すると、タググループ4.1のタグを付加するノードが存在しなくなったと判断し、ノード17自身をタググループ4.1に加入させる。

#### 【0365】

ツリー5.1のルートノードであるノード11が、ノード17をタググループ4.1に加入させる場合は、ノード11～17が、タググループ4.2のタグが付いたBPDUsに現用系フラグが付加させたのを確認し、クライアントより転送されてくるフレームに付加するタグを、タググループ4.1からタググループ4.2に切り替える際に、切替完了通知をツリー5.1のルートノードであるノード11に対して送信し、ノード11が、ノード11～17のすべてのノードから切替完了通知を受信すると、タググループ4.1のタグを付加するノードが存在しなくなったと判断し、ノード17に対してGVRPフレームを送信し、タググループ4.1に入れるよう命令する。

#### 【0366】

以上のようにして、ネットワークを停止させることなく、ノード17を追加することができる。以降ノードの追加の際は、同様の動作を繰り返す。ただし、上記説明文中のタググループ4.1とタググループ4.2は適宜入れ替わる。

#### 【0367】

次に、本実施の形態の効果について説明する。

#### 【0368】

従来、スパニングツリーに属するノードの追加および削除時は、全体もしくは一部のデータフレームの転送を停止させて、スパニングツリーを構築しなおすため、再構築中にネットワークが停止する場合があった。

#### 【0369】

本実施の形態では、構成変更前のスパニングツリーを運用したまま、新規追加

ノードを含めたスパニングツリーを生成し、新規スパニングツリーの安定後に利用するスパニングツリーを切り替えることにより、ネットワークを停止させずに、スパニングツリーに属するノードの追加および削除等のスパニングツリー再構成が可能である。

## 【0370】

またこの結果、輻輳発生の可能性を下げることが可能である。

## 【0371】

(第2の実施の形態)

以下、本発明の第2の実施の形態について、図面を参照して詳細に説明する。

## 【0372】

本発明の第2の実施の形態は、第1の実施の形態において、コストの計算時にリンク帯域の太さの代わりに、空き帯域の容量もしくは通過TCPフロー数、HTTPリクエスト数等を利用し、さらに、コストが変更された場合には、ノードの追加・削除の場合と同様に、現用系と予備系の遷移を行う点が異なる。なお、以降は空き帯域の容量をコストとして用いた場合について記述するが、通過TCPフロー数およびHTTPリクエスト数についても、特に断りのない限り同様に実現可能である。

## 【0373】

IEEE802.1DおよびIEEE802.1wでは、リンクのコストはリンク帯域の太さの逆数で決定されていた。つまり、負荷に応じて動的にコストを変化させることができなかった。

## 【0374】

本実施の形態においては、リンクのコストをリンクの空き帯域の逆数で決定することで、負荷に応じた動的なコスト変更を行う。

## 【0375】

図17は、第2の実施の形態におけるツリーセレクタ116の構成を示している。図17を参照すると、本発明の第2の実施の形態は、第1の実施の形態における図11のツリーセレクタにおいて、コスト参照タイマ1167、関数演算器1168、および、平滑化回路1169が追加されている点において異なる。

## 【0376】

メインコントローラ1164 $\alpha$ は、第1の実施における動作のほか、コスト参照タイマ1167より満了通知を受信すると、リソースモニタから、前回のコスト参照タイマ満了以降にリンクを流れたフレームの流量情報もしくはTCPフロー数またはHTTPリクエスト数を取得し、その流量、フロー数、もしくはリクエスト数を元にコストを計算し、予備用に登録されているツリーマネージャ（以下、予備系のツリーマネージャという）に通知する動作を行う。流量の場合は、流量およびリンク帯域の太さより、リンクの空き帯域を求め、前記リングの空き帯域の逆数をコストとして利用する。TCPフロー数およびHTTPリクエスト数の場合は、あらかじめ設定してある最大許容フロー数および最大許容リクエスト数と、実際にリンクを通過するTCPフロー数もしくはリクエスト数との差を取り、この差の逆数をコストとして利用する。

## 【0377】

メインコントローラ1164 $\alpha$ は、前記手段によりコストを算出した後、関数演算器1168にコストを渡して評価し、さらに関数演算器の評価結果を平滑化回路1169に渡して平滑化し、平滑化された結果の値を、予備系のツリーマネージャに対して送信する。

## 【0378】

関数演算器1168は、メインコントローラ1164 $\alpha$ より入力されたコスト値をパラメータとして、比例関数、ヒステリシス関数、および階段関数等、あらかじめ指定されている任意の関数を用いて、出力コスト値を決定し、メインコントローラ1164に返す、関数演算器1168の働きにより、状態遷移の振動を防止することが可能となる。コスト値の急激な変動を抑えコストを滑らかに変化させるためである。

## 【0379】

平滑化回路1169は、ローパスフィルタ等を用いて、あらかじめ保存してある前回の入力パラメータと、メインコントローラ1164 $\alpha$ より新たに渡された入力パラメータの平滑化を行い、結果をメインコントローラ1164 $\alpha$ に通知する。平滑化回路1169の働きにより、コストの急激な変動や、状態遷移の振動

を防止することが可能となる。

【0380】

コスト参照タイマ1167は、メインコントローラ1164 $\alpha$ より送信されたセット通知の受信から、あらかじめ設定された時間が経過すると、メインコントローラ1164 $\alpha$ に対してタイマ満了通知を送信する。

【0381】

図18は、本発明の第2の実施の形態の図17における、メインコントローラ1164Aの状態遷移を詳細に示した流れ図である。図18を参照すると、本発明の第2の実施の形態は、第1の実施の形態における図12において、状態1164Aおよび状態1164Bが追加されている点において異なる。

【0382】

状態1164Aは、現用系ツリーマネージャが、ツリーマネージャ1152であり、予備系ツリーマネージャがツリーマネージャ1151であり、ツリーマネージャ1152内のBPDUs送受信機11522のBPDUs送信機能が有効にされ、ツリーマネージャ1151内のBPDUs送受信機11512のBPDUs送信機能も有効にされている状態である。なお、BPDUs送受信機11512およびBPDUs送受信機11522のBPDUs受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

【0383】

状態1164Bは、現用系ツリーマネージャが、ツリーマネージャ1151であり、予備系ツリーマネージャがツリーマネージャ1152であり、ツリーマネージャ1151内のBPDUs送受信機11512のBPDUs送信機能が有効にされ、ツリーマネージャ1152内のBPDUs送受信機11522のBPDUs送信機能も有効にされている状態である。なお、BPDUs送受信機11512およびBPDUs送受信機11522のBPDUs受信機能は、送信機能の有効もしくは無効に関わらず、常に有効である。

【0384】

以下、図18を参照にして、状態11645を基点としてコスト計算の流れを示すが、この動作は状態11646を基点とした場合においても、同様に適用可

能である。

#### 【0385】

メインコントローラ1164 $\alpha$ は、状態11645に遷移した際、設定インターフェース118もしくはGVRP送受信機1162より、動的なコスト計算を利用する指定を受けた場合に、コスト参照タイマ1167をセットする。（状態11645）

メインコントローラ1164 $\alpha$ は、コスト参照タイマ1167からタイマ満了の通知を受信すると、リソースモニタ119より累積通過バイト数の情報を受信すると同時に、カウンタリセット通知を出して、リソースモニタ119の累積通過バイト数を0にリセットさせる。さらに、前記累積通過バイト数もしくはTCPフロー数またはHTTPリクエスト数よりコストを計算し、結果を関数演算器1168に渡す。

#### 【0386】

関数演算器1168は、メインコントローラ1164 $\alpha$ より入力された値を、設定された関数により評価し、結果をメインコントローラ1164 $\alpha$ に返却する。ここでは、比例関数が設定されており、ある入力値に対する出力値が同一になる例をとって解説する。

#### 【0387】

メインコントローラ1164 $\alpha$ は、関数演算器1168よりコストの評価結果を受け取ると、その値を平滑化回路1169に通知する。

#### 【0388】

平滑化回路1169は、設定にしたがって入力値をローパスフィルタ等によって平滑化し、結果をメインコントローラ1164 $\alpha$ に返却する。

#### 【0389】

メインコントローラ1164 $\alpha$ は、平滑化回路1169より平滑化完了後のコスト値を受信すると、このコスト値を予備系のツリーマネージャ1151に通知する。ツリーマネージャ1151は、このコスト情報を元に、スパンニングツリーを再計算し、計算の結果、トポロジに変更があるか、ないかを、メインコントローラ1164 $\alpha$ に通知する。（図18の状態1164A）

## 【0390】

メインコントローラ1164 $\alpha$ は、状態1164Aにおいて、再計算後のツリーが計算前のツリーと同一もしくは、あらかじめ設定された変化よりも変化の度合いが小さい場合は、状態11645に遷移し、コスト参照タイマを再セットする。図18の状態遷移図では、一例として、わずかでも変化がある場合は、状態11645には遷移しない設定であるとして説明している。（状態1164A）

## 【0391】

メインコントローラ1164 $\alpha$ は、状態1164Aにおいて、再計算後のツリーが計算前のツリーと異なる場合で、かつ、その変化があらかじめ設定された変化より大きいものであるとき、安定タイマ1165をセットし、タイマ満了後に状態1164Bに遷移する。図18の状態遷移図では、わずかでも変化がある場合には、1164Bに遷移する設定であるとして説明している。（状態1164A）

## 【0392】

メインコントローラ1164 $\alpha$ は、安定タイマ1165の満了通知を受信すると、現用に登録されているツリーマネージャ1152と、予備用に登録されているツリーマネージャ1151を入れ替え、新しく現用をツリーマネージャ1151、予備用をツリーマネージャ1152とする。さらに、GVRP送受信機1162を通じて、新ツリーのルートノードに対して、利用タググループ変更通知を送信する。この利用タググループ変更通知の内容は、BPDUsに反映され、全ノードに伝達される。その後、安定タイマ1165を作動させ、状態11646に遷移する。（状態1164B）

## 【0393】

メインコントローラ1164 $\alpha$ は、安定タイマ1165の満了通知を受信すると、ツリーマネージャ1152に対して新規コストの通知を行い、新たに計算したコスト情報を元に、スパンニングツリーを再計算させる。また、コスト参照タイマ1167をスケジュールする。（状態11646）

## 【0394】

次に図7を利用して、本実施の形態において、ノード15からノード13への

転送経路が変更される場合の、スパンニングツリー切り替え動作について、具体例を用いて詳細に説明する。

【0395】

図7を参照すると、本動作例では、ノード11～16、クライアント91～96、双方向リンク81～86、および、双方向リンク21～28を含む。

【0396】

ノード11にはリンク81によりクライアント91が、ノード12にはリンク82によりクライアント92が、ノード13にはリンク83によりクライアント93が、ノード14にはリンク84によりクライアント94が、ノード15にはリンク85によりクライアント95が、ノード16にはリンク26によりクライアント96が、それぞれ接続されている。

【0397】

ノード11とノード12の間はリンク21で、ノード12とノード13の間はリンク22で、ノード13とノード14の間はリンク23で、ノード11とノード15の間はリンク24で、ノード15とノード16の間はリンク25で、ノード16とノード14の間はリンク26で、ノード12とノード15の間はリンク27で、ノード13とノード16の間はリンク28で、それぞれ接続されている。

【0398】

ノード11～16の全ポートが所属するタググループが2つすでに設定されており、1つめのタググループをタググループ41、2つめのタググループをタググループ42と呼ぶ。

【0399】

ノード11～16には、独立して動作するスパンニングツリー回路が2つ存在し、タググループ41上で動作するスパンニングツリーをツリー51、タググループ42上で動作するスパンニングツリーをツリー52と呼ぶ。

【0400】

図7において、ツリー41は、ノード13をルートノードとし、すべてのリンクに等しく10に設定された初期コストを用いて、すでに安定しているとする。

ツリー51のBPDUsにはタググループ41のタグが、ツリー52のBPDUsにはタググループ42のタグがつく。

【0401】

現在、クライアントからノード11～16に送信されたデータには、現用系であるタググループ41のタグが付加されている。このタグが付加されたデータは、ツリー51に沿って転送されている。

【0402】

初期状態では、クライアント91～96のうち、どのクライアントもデータ転送を行っていない。

【0403】

各ノードは、一定時間ごとに、Hello Timeで規定される周期でBPDUsを送受信することにより、BPDUsの状態を確認している。このフレームには、現用系と予備系の識別フラグがあり、現在は現用系であるタググループ41のタグが付加されたBPDUsのみに、現用系のフラグがつき、予備系であるタググループ42のタグが付加されたBPDUsには、現用系を示すフラグは付加されない。

【0404】

各ノードは、設定インターフェース118もしくはGVRP送受信機1162より、すでに動的なコスト計算を利用する指定を受けているため、コスト参照タイムが満了するごとに、前回のタイム満了以降にリンクを流れたフレームの流量を参照し、コストの再計算および、予備系ツリーを用いたスパニングツリーの再計算を行う。

【0405】

ここで、クライアント95からクライアント93、および、クライアント96からクライアント93との間で、データ転送を開始したとする。

【0406】

転送初期においては、クライアント95からクライアント93へのデータは、ツリー51を利用し、リンク85、25、28、83を経由して転送される。また、クライアント96からクライアント93へのデータも、ツリー51を利用し

、リンク86、28、83を経由して転送される。

#### 【0407】

データ転送から一定時間が経過すると（各ノードのコスト参照タイマ1167が満了すると）、リンク21～28の空き容量をもとに、ツリー52におけるリンク21～28のコストが再計算され、関数演算器および平滑化回路にかけられる。ここで、リンク28の空き帯域は少なくなっているため、ノード16によりツリー52のリンク28のコストは15に、リンク25の空き帯域は、リンク28ほどではないが少なくなっているため、ノード15によりツリー52のリンク25のコストは12に、それぞれ変更されたとする。このとき、現在使用している、ツリー51のコストは変更されない。

#### 【0408】

ノード16は、コスト変更を検知し、変更後のコストを利用して作成したBPDU（図5のRoot path Cost 22056の値が変更される）を、隣接するノード13、ノード14、ノード15に対して、それぞれ送信する。このBPDUにはタググループ42のタグが付加される。

#### 【0409】

ノード15も同様に、コスト変更を検知し、変更後のコストを利用して作成したBPDUを、隣接するノード11、ノード12、ノード16に対して、それぞれ送信する。

#### 【0410】

ノード15は、ノード16からコスト15が付加されたBPDUを受信すると、リンク25のコストも加算し、リンク25、リンク28経由でノード13に到達するために、コスト27がかかるのことを認識する。

#### 【0411】

その後ノード15は、ノード12から、コスト10が付加されたBPDUを受信すると（ノード12は、ルートノード（ノード13）までのコストを、定期的に隣接ノード（ノード11、およびノード15）に送信する）、リンク27のコストも加算し、リンク27、リンク22経由でノード13に到達するために、コスト20がかかるのことを認識する。このコストは、リンク25経由のコストより

も小さいため、ノード15は停止ポートをリンク27側からリンク25側に切り替え、リンク27およびリンク22を利用して、リンク13に到達するツリーを形成する。リンク27、22経由のツリーを作成するときにもスパニングツリープロトコルが使用される。そして、安定タイマ1165を起動し、ツリーが安定するのを待つ。

#### 【0412】

ここで、ツリー52が、ノード13をルートノードとして安定したとする。

#### 【0413】

ツリー51のルートノードであるノード13は、図13の安定タイマ1165が満了すると、ツリー形成が安定したと判断し、新ツリーとなるツリー52のルートノードであるノード13に対して、転送に利用する現用系ツリーをツリー51からツリー52に切り替えるよう命令する、タググループ変更メッセージを送信する。その後、安定タイマ1165を作動させる。

#### 【0414】

ツリー52のルートノードであるノード13は、ツリー切替メッセージを受信すると、旧現用系ツリーであるツリー41のルートであるノード13に対して、転送に利用する現用系ツリーをツリー51からツリー52に切り替えるよう命令する、タググループ変更メッセージを送信する。この利用タググループ変更通知の内容は、ノード13から送信される、ツリー51およびツリー52の各BPDUに反映され、全ノードに伝達される。

#### 【0415】

ノード13は、旧現用系ルートノードへのタググループ変更メッセージの送信を完了させると、自ノードがクライアント93より受信して、ネットワーク内に送信するフレームの転送に利用するツリーを、今まで利用していたツリー51からツリー52へ切り替える。切り替えが完了すると、クライアント93からクライアント95に向けて送信されていたフレームは、リンク83、リンク22、リンク27、リンク85を経由して、クライアント95に転送されるようになる。

#### 【0416】

このようにして、クライアント93からクライアント95、および、クライア

ント93からクライアント96に向けて送信されるフレームの転送経路は分散され、リンク28の混雑が解消される。

## 【0417】

以降、コスト参照タイマが満了するごとに、リンクの空き帯域に基づくコスト計算によってスパニングツリーが再計算され、周期的に空き帯域をコストに反映する動的な経路変更が行われる。この結果、各リンクのトラフィックが分散され、リンクの負荷を分散し、輻輳を防止することができる。

## 【0418】

次に、本実施の形態の効果について説明する。

## 【0419】

従来、リンク容量を用いてコストを計算し、スパニングツリー構築時の経路選択に利用しているため、トラフィックに応じた動的な負荷分散のための経路変更ができなかった。

## 【0420】

本実施の形態では、リンクコストを、空き帯域やサーバ負荷等の動的情報によって計算することにより、トラフィックの負荷分散が可能である。

## 【0421】

また従来、コストをトラフィック状況に応じて動的に変化させようとすると、局所的もしくはネットワーク全体で、データフレームの転送を停止させてスパニングツリーを構築しなおして経路を変更するため、再構築中にネットワークが停止する場合があった。

## 【0422】

本実施の形態では、変更前のツリーを運用したまま、コスト変更後のツリーを生成し、新規ツリーの安定後に利用するツリーを切り替えることにより、経路変更にともなうスパニングツリー再構成のために、ネットワークを停止させることなく負荷分散が可能である。

またこの結果、輻輳発生の可能性を下げることが可能である。

## 【0423】

(第3の実施の形態)

以下、本発明の第3の実施の形態について、図面を参照して詳細に説明する。

【0424】

本発明の第3の実施の形態は、第2の実施の形態において、コストが変更されたかされなかったかに関わらず、現用系と予備系の遷移を行う点が異なる。なお、以降は空き帯域の容量をコストとして用いた場合について記述するが、通過TCPフロー数およびHTTPリクエスト数についても、特に断りのない限り同様に実現可能である。

【0425】

図19を参照すると、本発明の第3の実施の形態は、第2の実施の形態における図18において、状態1164Aと状態1164Bの間の遷移が発生しないこと、ならびに、状態11643と状態11644の間の遷移および、状態11647と状態11648の間の遷移が、安定タイマ1165の満了によって起こるのではなく、BPDUs中の現用系フラグの切り替え検出によって発生する点が異なる。

【0426】

第3の実施の形態におけるメインコントローラ1164βは、リソースモニタ119より新たにネットワークに接続されたことの通知を受信すると、ツリーマネージャ1151もしくはツリーマネージャ1152からの現用系通知の到着を待つ。ツリーマネージャ1151もしくはツリーマネージャ1152よりBPDUsに含まれる現用系通知を受信すると、前記通知において指定されたツリーマネージャ1151もしくはツリーマネージャ1152を現用に、予備用に指定されたツリーマネージャ1151もしくは1152を予備用に設定し、状態11642もしくは状態11649に遷移する。ここでは、状態11642に遷移した場合を例に解説するが、以後の解説は状態11649に遷移した場合においても同様である。（図19の状態11641）

【0427】

メインコントローラ1164βは、ツリーマネージャ1151を現用、ツリーマネージャ1152を予備用に設定する。さらに、前記各ツリーマネージャ1151および1152に対して、BPDUs送信停止命令を出す。（図19の状態1

1642)

## 【0428】

メインコントローラ1164βは、状態11642において、もし設定インターフェース118よりノードの追加要求を受信した場合は、状態11643に遷移する。また、もしツリーマネージャ1151もしくはツリーマネージャ1152より現用系通知を受信し、現用系と予備系の関係に変更があった場合には、状態11649に遷移する。（図19の状態11642）

## 【0429】

メインコントローラ1164βは、ツリーマネージャ1152に対してリンクupの通知を行い、同時に、ツリーマネージャ1152に対してBPDUの送信許可を行う。さらに、安定タイマ1165を作動させる。（図19の状態11643）

メインコントローラ1164βは、状態11643において、ツリーマネージャ1151もしくはツリーマネージャ1152よりBPDUに含まれる現用系通知を受信し、もし現用系と予備系の関係に変更が生じた場合には、現用に登録されているツリーマネージャ1151と、予備用に登録されているツリーマネージャ1152を入れ替え、新しく現用をツリーマネージャ1152、予備用をツリーマネージャ1151とする。その後、安定タイマ1165を作動させる。（図19の状態11644）

## 【0430】

メインコントローラ1164βは、安定タイマ1165よりタイマ満了通知を受信すると、ツリーマネージャ1151に対してリンクupの通知を行い、同時に、ツリーマネージャ1152に対してBPDUの送信許可を行う。通常は、この状態で安定する。（図19の状態11645）

## 【0431】

状態11645において、もし現用であるツリーマネージャ1152がルートノードとなっている場合、このノードのメインコントローラ1164βは、安定タイマ1165を動作させ、安定タイマ1165よりタイマ満了通知を受信すると、現用に登録されているツリーマネージャ1152と、予備用に登録されてい

るツリーマネージャ1151を入れ替え、新しく現用をツリーマネージャ1151、予備用をツリーマネージャ1152とする。さらに、GVRP送受信機1162を通じて、新ツリーのルートノードに対して、利用タググループ変更通知を送信する。この利用タググループ変更通知の内容は、BPDUsに反映され、全ノードに伝達される。（図19の状態11645）

#### 【0432】

前記ツリーマネージャの入れ替えは、予備用であるツリーマネージャ1151が行っても良い。この場合、状態11645において、もし予備用であるツリーマネージャ1151がルートノードとなっている場合、ノードのメインコントローラ1164は、安定タイマ1165を動作させ、安定タイマ1165よりタイマ満了通知を受信すると、現用に登録されているツリーマネージャ1152と、予備用に登録されているツリーマネージャ1151、予備用をツリーマネージャ1152とする。さらに、GVRP送受信機1162を通じて、旧現用ツリーのルートノードに対して、利用タググループ変更通知を送信する。この利用タググループ変更通知の内容は、BPDUsに反映され、全ノードに伝達される。（図19の状態11645）

#### 【0433】

メインコントローラ1164βは、状態11645において、ツリーマネージャ1151もしくはツリーマネージャ1152よりBPDUsに含まれる現用系通知を受信し、もし現用系と予備系の関係に変更が生じた場合には、状態11646に遷移する。そして、現用に登録されているツリーマネージャ1152と、予備用に登録されているツリーマネージャ1151を入れ替え、新しく現用をツリーマネージャ1151、予備用をツリーマネージャ1152とする。（図19の状態11645）

#### 【0434】

メインコントローラ1164βは、状態11645において、もし設定インターフェース118より、ノード削除要求を受信した場合は、状態11644に遷移する。（図19の状態11645）

#### 【0435】

メインコントローラ1164 $\beta$ は、ツリーマネージャ1151に対して、接続されている全リンクリンクdownの通知を行い、同時に、ツリーマネージャ1151に対してBPDUsの送信停止命令を出す。（図19の状態11644）

#### 【0436】

メインコントローラ1164 $\beta$ は、状態11644において、ツリーマネージャ1151もしくはツリーマネージャ1152よりBPDUsに含まれる現用系通知を受信し、もし現用系と予備系の関係に変更が生じた場合には、状態11643に遷移し、現用に登録されているツリーマネージャ1152と、予備用に登録されているツリーマネージャ1151を入れ替え、新しく現用をツリーマネージャ1151、予備用をツリーマネージャ1152とする。その後、安定タイマ1165を作動させる。（図19の状態11643）

#### 【0437】

メインコントローラ1164 $\beta$ は、安定タイマ1165よりタイマ満了通知を受信すると、ツリーマネージャ1151に対してリンクdownの通知を行い、同時に、ツリーマネージャ1151に対してBPDUsの受信停止命令を出す。さらに、無条件で状態11641に遷移し、ノード切離しまで待機する。（図19の状態11642）

#### 【0438】

メインコントローラ1164 $\beta$ は、設定インターフェース118もしくはGVRP送受信機1162より、動的なコスト計算を利用する指定を受けた場合には、利用タググループ変更通知の受信から一定時間経過後に満了となるよう、コスト参照タイマ1167をセットする。（図19の状態11645）

#### 【0439】

メインコントローラ1164 $\beta$ は、コスト参照タイマ1167からタイマ満了の通知を受信すると、リソースモニタ119より累積通過バイト数の情報を受信すると同時に、カウンタリセット通知を出して、リソースモニタ119の累積通過バイト数を0にリセットさせる。さらに、前記累積通過バイト数もしくはTCPフロー数またはHTTPリクエスト数よりコストを計算し、予備系であるツリーマネージャ1151通知する。ツリーマネージャ1151は、新たに計算した

コスト情報を元に、スパンニングツリーを再計算し、計算の結果、トポロジに変更があるか、ないかを、メインコントローラ1164βに通知する。そして、無条件で状態11645に遷移する。（図19の状態1164A）

#### 【0440】

図14および図15を参照して、本実施の形態において、ノード17を追加する場合の動作について、具体例を用いて詳細に説明する。

#### 【0441】

図14を参照すると、本動作例では、ノード11～17、およびリンク21～30を有する。ただし、ノード17、リンク29およびリンク30は、初期状態では接続されていない。

#### 【0442】

図15を参照すると、本動作例では、ノード11～17、およびリンク21～30を有する。

#### 【0443】

初期状態でノード11～16の全ノード、全ポートが所属するタググループが2つすでに設定されており、1つめのタググループをタググループ41、2つめのタググループをタググループ42と呼ぶ。

#### 【0444】

なお、基本的には全ノード、全ポートを2つのタググループに加入させるが、一部のポートもしくはノードのみで構成させるタググループを作成しても良い。以降、全ノード、全ポートが2つのタググループに加入するものとして説明する。

#### 【0445】

ノード11～16には、独立して動作するスパンニングツリー回路が2つ存在し、タググループ41上で動作するスパンニングツリーをツリー51、タググループ42上で動作するスパンニングツリーをツリー52と呼ぶ。

#### 【0446】

スパンニングツリーは必ず2系統作成するが、タググループは必ずしも2つ作成する必要は無い。前記タググループ41のみ設定してタググループ42を利用せ

ず、タググループ4 1 上で動作するスパンニングツリーをツリー5 1、タググループに属さないで動作するスパンニングツリーをツリー5 2と呼ぶこともできる。また逆に、前記タググループ4 2のみ設定してタググループ4 1を利用せず、タググループに属さないで動作するスパンニングツリーをツリー5 1、タググループ4 2上で動作するスパンニングツリーをツリー5 2と呼ぶこともできる。

#### 【0447】

ここでは、特にタググループ4 1とタググループ4 2の両方を用いた場合について説明を行うが、タググループ4 1及びタググループ4 2の両方を用いた場合の動作は、タググループ4 1のみ、もしくはタググループ4 2のみを用いた場合においても、同様に適用可能である。

#### 【0448】

図14では、ツリー5 1は太線で表され、ノード1 1をルートノードとして安定しているとする。

#### 【0449】

ツリー5 1のBPDUsにはタググループ4 1のタグが、ツリー5 2のBPDUsにはタググループ4 2のタグがつく。

#### 【0450】

現在、クライアントからノード1 1～1 6に送信されたデータには、タググループ4 1のタグが付加されている。このタグが付加されたデータは、ツリー5 1に沿って転送されている。

#### 【0451】

ノード1 7は、リンク2 9およびリンク3 0と接続されると、どのタググループにも加入せず、BPDUsの受信を開始する。

#### 【0452】

ノード1 7は、各タググループのBPDUsを受信すると、現時点での現用系がタググループ4 1で、現時点での予備系がタググループ4 2であると確認する。そして、自ノードをタググループ4 2にのみ参加させてBPDUsの送受信を行い、タググループ4 1ではBPDUsは受信のみを行い、送信を行わないように設定する。

## 【0453】

ノード17が追加されると、タググループ42のメンバに変更が発生するため、スパニングツリープロトコルにより、ツリー52を更新する動作が開始される。タググループ41のメンバには変更が無いため、ツリー51は更新されない。

## 【0454】

クライアントから送信されるフレームには、今までどおりタググループ41のタグが付加され、ツリー51に沿って転送され続ける。

## 【0455】

現用系ツリーであるツリー51のルートノードとなっているノード11は、安定タイマが満了すると、ツリー51のルートノードであるノード11に対して利用タググループ変更通知を送信し、ツリー51を予備系に遷移させるよう命令する。そして、タググループ42のフラグを付け、ツリー52のために送信するBPDUsに、現用系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

## 【0456】

ノード11は、現用系フラグの付加状況の変更を感じし、ツリー51を予備系に遷移させ、タググループ41のフラグを付け、ツリー51のために送信するBPDUsに、予備系フラグを立てる。BPDUsは、各ノードで転送されながら、全ノードに伝播する。

## 【0457】

ノード11～17は、タググループ42のタグが付いたBPDUsに現用系フラグが付加させたのを確認し、クライアントより転送されてくるフレームに付加するタグを、タググループ41からタググループ42に切り替える。このタグが付加されたフレームは、ツリー52に沿って転送される。

## 【0458】

以上の切り替えが完了してしばらく経つと、ツリー51を流れるフレームはなくなる。

## 【0459】

ノード17は、利用タググループ変更通知の送信から一定時間経過後に、タグ

グループ41のタグを付加するノードが存在しなくなつたと判断し、自ノードをタググループ41に加入させ、次のトポロジ変更に備える。

#### 【0460】

以上のようにして、ネットワークを停止させることなく、ノード17を追加することができた。以降ノードの追加の際は、同様の動作を繰り返す。ただし、上記説明文中のタググループ41とタググループ42は入れ替わる。

#### 【0461】

次に図7を利用して、本実施の形態において、ノード15からノード13への転送経路が変更される場合の、スパンギングツリー切り替え動作について、具体例を用いて詳細に説明する。

#### 【0462】

図7を参照すると、本動作例では、ノード11～16、クライアント91～96、双方向リンク81～86、および、双方向リンク21～28を含む。

#### 【0463】

ノード11にはリンク81によりクライアント91が、ノード12にはリンク82によりクライアント92が、ノード13にはリンク83によりクライアント93が、ノード14にはリンク84によりクライアント94が、ノード15にはリンク85によりクライアント95が、ノード16にはリンク26によりクライアント96が、それぞれ接続されている。

#### 【0464】

ノード11とノード12の間はリンク21で、ノード12とノード13の間はリンク22で、ノード13とノード14の間はリンク23で、ノード11とノード15の間はリンク24で、ノード15とノード16の間はリンク25で、ノード16とノード14の間はリンク26で、ノード12とノード15の間はリンク27で、ノード13とノード16の間はリンク28で、それぞれ接続されている。

#### 【0465】

ノード11～16の全ポートが所属するタググループが2つすでに設定されており、1つめのタググループをタググループ41、2つめのタググループをタグ

グループ4 2と呼ぶ。

【0466】

ノード11～16には、独立して動作するスパニングツリー回路が2つ存在し、タググループ4 1上で動作するスパニングツリーをツリー5 1、タググループ4 2上で動作するスパニングツリーをツリー5 2と呼ぶ。

【0467】

ツリー4 1は太線で表され、ノード13をルートノードとし、すべてのリンクに等しく10に設定された初期コストを用いて、すでに安定しているとする。ツリー5 1のBPDUsにはタググループ4 1のタグが、ツリー5 2のBPDUsにはタググループ4 2のタグがつく。

【0468】

現在、クライアントからノード11～16に送信されたデータには、現用系であるタググループ4 1のタグが付加されている。このタグが付加されたデータは、ツリー5 1に沿って転送されている。

【0469】

初期状態では、クライアント91～96のうち、どのクライアントもデータ転送を行っていない。

【0470】

各ノードは、一定時間ごとに、HELLOフレームを送信し、BPDUsの状態を確認している。このフレームには、現用系と予備系の識別フラグがあり、現在は現用系であるタググループ4 1のタグが付加されたBPDUsのみに、現用系のフラグがつき、予備系であるタググループ4 2のタグが付加されたBPDUsには、現用系を示すフラグは付加されない。

【0471】

各ノードは、設定インターフェース118もしくはGVRP送受信機1162により、すでに動的なコスト計算を利用する指定を受けているため、コスト参照タイムが満了するごとに、前回のタイム満了以降にリンクを流れたフレームの流量を参照し、コストの再計算および、予備系ツリーを用いたスパニングツリーの再計算を行う。

## 【0472】

ここで、クライアント95からクライアント93、および、クライアント96からクライアント93との間で、データ転送を開始したとする。

## 【0473】

転送初期においては、クライアント95からクライアント93へのデータは、ツリー51を利用し、リンク85、25、28、83を経由して転送される。また、クライアント96からクライアント93へのデータも、ツリー51を利用し、リンク86、28、83を経由して転送される。

## 【0474】

データ転送から一定時間が経過すると、リンク21～28の空き容量をもとに、ツリー52におけるリンク21～28のコストが片側ずつ再計算される。ここで、リンク28の空き帯域は少なくなっているため、ノード16によりツリー52のリンク28のコストは15に、リンク25の空き帯域は、リンク28ほどではないが少なくなっているため、ノード15によりツリー52のリンク25のコストは12に、それぞれ変更されたとする。このとき、現在使用している、ツリー51のコストは変更されない。

## 【0475】

ノード16は、コスト変更を検知し、変更後のコストを利用して作成したBPDUを、隣接するノード13、ノード14、ノード15に対して、それぞれ送信する。

## 【0476】

ノード15も同様に、コスト変更を検知し、変更後のコストを利用して作成したBPDUを、隣接するノード11、ノード12、ノード16に対して、それぞれ送信する。

## 【0477】

ノード15は、ノード16からコスト15が付加されたBPDUを受信すると、リンク25のコストも加算し、リンク25、リンク28経由でノード13に到達するために、コスト27がかかるのことを認識する。

## 【0478】

その後ノード15は、ノード12から、コスト10が付加されたBPDUsを受信すると、リンク27のコストも加算し、リンク27、リンク22経由でノード13に到達するために、コスト20がかかるのを認識する。このコストは、リンク25経由のコストよりも小さいため、ノード15は停止ポートをリンク27側からリンク25側に切り替え、リンク27およびリンク22を利用して、リンク13に到達するツリーを形成する。

#### 【0479】

ツリー51のルートノードであるノード13において起動していたトポロジ安定タイマが満了すると、旧現用系ツリーであるツリー41のルートであるノード13に対して、転送に利用する現用系ツリーをツリー51からツリー52に切り替えるよう命令する、タググループ変更メッセージを送信する。この利用タググループ変更通知の内容は、ノード13から送信される、ツリー51およびツリー52の各BPDUsに反映され、全ノードに伝達される。

#### 【0480】

ノード13は、旧現用系ルートノードへのタググループ変更メッセージの送信を完了させると、自ノードがクライアント93より受信して、ネットワーク内に送信するフレームの転送に利用するツリーを、今まで利用していたツリー51からツリー52へ切り替える。切り替えが完了すると、クライアント93からクライアント95に向けて送信されていたフレームは、リンク83、リンク22、リンク27、リンク85を経由して、クライアント95に転送されるようになる。

#### 【0481】

このようにして、クライアント93からクライアント95、および、クライアント93からクライアント96に向けて送信されるフレームの転送経路は分散され、リンク28の混雑が解消される。

#### 【0482】

以降、コスト参照タイマが満了するごとに、リンクの空き帯域に基づくコスト計算によってスパンニングツリーが再計算され、周期的に空き帯域をコストに反映する動的な経路変更が行われる。この結果、各リンクのトラフィックが分散され、リンクの負荷を分散し、輻輳を防止することができる。

【0483】

次に、本実施の形態の効果について説明する。

【0484】

従来、リンク容量を用いてコストを計算し、スパニングツリー構築時の経路選択に利用しているため、トラフィックに応じた動的な負荷分散のための経路変更ができなかった。

【0485】

本実施の形態では、リンクコストを、空き帯域やサーバ負荷等の動的情報によって計算することにより、トラフィックの負荷分散が可能である。

【0486】

また従来、コストをトラフィック状況に応じて動的に変化させるたびに、局所的もしくはネットワーク全体で、データフレームの転送を停止させてスパニングツリーを構築しなおして経路を変更するため、再構築中にネットワークが停止する場合があった。

【0487】

本実施の形態では、変更前のツリーを運用したまま、コスト変更後のツリーを生成し、新規ツリーの安定後に利用するツリーを切り替えることにより、経路変更にともなうスパニングツリー再構成のために、ネットワークを停止させることなく負荷分散が可能である。

またこの結果、輻輳発生の可能性を下げることが可能である。

【0488】

(第4の実施の形態)

以下、本発明の第4の実施の形態について、図面を参照して詳細に説明する。

【0489】

本発明の第4の実施の形態は、第1の実施の形態において、宛先ノード別に利用するタグおよびスパニングツリーを切り替え、さらに、宛先のノードをルートノードに設定する場合に対応する。

【0490】

I E E E 8 0 2. 1 D および I E E E 8 0 2. 1 w が動作しているネットワー

クにおいてフレームを送信した場合、宛先への最低コスト経路が必ずしも選択されるとは限らず、使用されないリンクが出現するほか、ルートノードに負荷が集中する、ルートノード障害時に長時間ネットワークが停止するなどの問題があった。

#### 【0491】

本実施の形態においては、宛先がルートノードとなるツリーを用いてフレームを転送することで、宛先への最低コストによるフレーム伝送のほか、リンク利用率の向上、さらにはルートノード障害への耐性強化を行う。

#### 【0492】

図20を参照すると、本発明の第4の実施の形態は、第1の実施の形態における図8において、ツリーマネージャ1151が、ネットワークに存在するノードの数だけ設置されている点において異なる。

#### 【0493】

ツリーマネージャ1151は、本発明の第1の実施の形態における図8に示すツリーマネージャ1151と同様の機能を有する。

#### 【0494】

ツリーマネージャ1152および、ツリーマネージャ1153は、ツリーマネージャ1151と同様のツリーマネージャである。以後、ツリーマネージャ1151～ツリーマネージャ1153を代表してツリーマネージャ1151を用いて記述を行うが、ツリーマネージャ1151についての記述は、特に断りのない限りツリーマネージャ1152～ツリーマネージャ1153においても同様に適用可能である。

#### 【0495】

ツリーマネージャはツリーセレクタ116によって、ネットワーク内に存在するノードの数、もしくは、サブネット等を区切り階層化している場合は、同一階層内に存在するノードの数だけ作成される。したがって、ツリーマネージャは、1個以上無限大まで増える可能性があるが、図20においては前記ツリーマネージャを、ツリーマネージャ1151～1153としてまとめて表現する。

#### 【0496】

図20のツリーセレクタ116は、本発明の第1の実施の形態における図8に示すツリーセレクタ116の機能のほか、ネットワークもしくは階層内に新規ノードの存在を検出した場合に、新規のツリーマネージャを生成する機能を有する。また、前記新規ノードの検出を、他のノードに対して通知する機能、および、他のノードから新規ノードの検出通知を受信し、ツリーマネージャを生成する機能を有する。さらに、ノードの削除を検出して、ツリーマネージャを削除する機能、前記削除ノードの検出を他のノードに対して通知する機能、および、他のノードからノードの削除通知を受信し、ツリーマネージャを削除する機能を有する。

## 【0497】

図21は、本実施の形態の図20における、タグをキーとして出力ポートを決定する、フォワーディングテーブル114の構成例である。

## 【0498】

タグフィールド1141は、サーチのインデックスになるフィールドであり、このフィールドの情報と、受信したフレームのタグに書かれている内容が一致するかどうかを調べる。

## 【0499】

出力ポート1142は、受信したフレームのタグに書かれている内容とタグにフィールド1141の内容が一致した場合に、前記フレームをどのポートに転送すべきか記述するフィールドである。

## 【0500】

なお、本実施の形態は、本動作例に示すような、タグの内容によって転送先ポートを決定するタグフォワーディングを行う場合のみならず、従来より行われている、MACアドレスによって転送先を決定する通常のMACアドレス転送においても、同様に適用可能である。この場合は、出力ポートフィールド1142に、対応するMACアドレスに対応する複数のポートが記載される場合もある。

## 【0501】

図22は、本実施の形態の図20における、宛先MACアドレスをキーとして挿入するタグを決定する、タグテーブル117の構成例である。

## 【0502】

宛先MACアドレス1171は、サーチのインデックスになるフィールドであり、このフィールドの情報と、受信したフレームの宛先MACアドレスフィールド、すなわち、MAC DAフィールドに書かれている内容が一致するかどうか調べ、一致した場合には、前記受信フレームに対して、挿入タグフィールド1172に記載のタグを挿入する。

## 【0503】

挿入タグフィールド1172は、宛先MACアドレスフィールド1171に対する、挿入すべきタグが記載されるフィールドである。本実施例では、宛先ノードIDが記載されており、このIDが、タグとしてフレームに挿入される。

## 【0504】

図23は、ノード11をルートノードとするスパニングツリーの構成図である、ツリー61の構成図である。ツリー61は、ノード11のプライオリティ値を、ノード12～ノード16の各ノードよりも小さい値に設定して作成される。ツリー61は、ノード11に向かうフレームの送信および、ノード11よりノード12～ノード16の各ノードに対して、ブロードキャストフレームを送信する場合に利用される。

## 【0505】

図24は、ノード12をルートノードとするスパニングツリーの構成図である、ツリー62の構成図である。ツリー62は、ノード12のプライオリティ値を、ノード11およびノード13～ノード16の各ノードよりも小さい値に設定して作成される。ツリー62は、ノード12に向かうフレームの送信および、ノード12よりノード11およびノード13～ノード16の各ノードに対して、ブロードキャストフレームを送信する場合に利用される。

## 【0506】

図25は、ノード13をルートノードとするスパニングツリーの構成図である、ツリー63の構成図である。ツリー63は、ノード13のプライオリティ値を、ノード11～ノード12およびノード14～ノード16の各ノードよりも小さい値に設定して作成される。ツリー63は、ノード13に向かうフレームの送信

および、ノード13よりノード11～ノード12およびノード14～ノード16の各ノードに対して、ブロードキャストフレームを送信する場合に利用される。

#### 【0507】

図26は、ノード14をルートノードとするスパンニングツリーの構成図である、ツリー64の構成図である。ツリー64は、ノード14のプライオリティ値を、ノード11～ノード13およびノード15～ノード16の各ノードよりも小さい値に設定して作成される。ツリー64は、ノード14に向かうフレームの送信および、ノード14よりノード11～ノード13およびノード15～ノード16の各ノードに対して、ブロードキャストフレームを送信する場合に利用される。

#### 【0508】

図27は、ノード15をルートノードとするスパンニングツリーの構成図である、ツリー65の構成図である。ツリー65は、ノード15のプライオリティ値を、ノード11～ノード14およびノード16の各ノードよりも小さい値に設定して作成される。ツリー65は、ノード15に向かうフレームの送信および、ノード15よりノード11～ノード14およびノード16の各ノードに対して、ブロードキャストフレームを送信する場合に利用される。

#### 【0509】

図28は、ノード16をルートノードとするスパンニングツリーの構成図である、ツリー66の構成図である。ツリー66は、ノード16のプライオリティ値を、ノード11～ノード15よりも小さい値に設定して作成される。ツリー66は、ノード16に向かうフレームの送信および、ノード16よりノード11～ノード15の各ノードに対して、ブロードキャストフレームを送信する場合に利用される。

#### 【0510】

次に、図23～図28を利用して、ノード13が、ノード11～12およびノード14～16すでに構成されているネットワークに、新たに追加された場合の、ツリー63の作成動作について説明する。

#### 【0511】

前記ネットワークにノード13が追加されると、ノード13は隣接ノードより

送信されるBPDUsを受信して、新たに検出した識別タグごとにツリーマネージャを生成する。この例では、ノード11～12およびノード14～16の各ノードをルートノードとする、5個のツリーマネージャが生成される。

#### 【0512】

次に、ノード13は、ノードIDよりタグIDを生成し、自ノードのプライオリティ値を低く設定したツリーマネージャを生成し、そのツリーマネージャから出力されるBPDUsに対して、前記タグIDを付加して送信する。ここで、タグIDは43であったとする。

#### 【0513】

ノード12およびノード16は、タグIDが43のBPDUsを新たに受信し、ツリーマネージャを生成してから、タグID43を付加したBPDUsを隣接ノードに対して送信する。

#### 【0514】

以上のBPDUs送信動作が繰り返されることにより、ツリー63が完成する。

#### 【0515】

次に、図23～図28を参照して、前記の各図におけるノード11～ノード16の各ノードが、ノード11～ノード16の各ノードにフレームを送信する場合の手順について述べ、送信したフレームが最低コスト経路で宛先へ届けられること、並びに、リンク資源の負荷分散が行われていることを示す。なお、各リンクのコストは等しく、すでに各図におけるツリー61～ツリー66の各ツリーは構成が完了し、トポロジが安定しているとする。

#### 【0516】

ノード12～ノード16の各ノードから、ノード11へフレームを送信する場合は、ツリー61を使用する。例えば、ノード15からノード11へフレームを送信する場合は、ノード15は、データフレームにツリー61の識別タグであるタグID41を付加して送信する。

#### 【0517】

ノード11およびノード13～ノード16の各ノードから、ノード12へフレームを送信する場合は、ツリー62を使用する。例えば、ノード14からノード

12へフレームを送信する場合は、ノード14は、データフレームにツリー62の識別タグであるタグID42を付加して送信する。

【0518】

ノード11～ノード12およびノード14～ノード16の各ノードから、ノード13へフレームを送信する場合は、ツリー63を使用する。例えば、ノード11からノード13へフレームを送信する場合は、ノード11は、データフレームにツリー63の識別タグであるタグID43を付加して送信する。

【0519】

ノード11～ノード13およびノード15～ノード16の各ノードから、ノード14へフレームを送信する場合は、ツリー64を使用する。例えば、ノード12からノード14へフレームを送信する場合は、ノード12は、データフレームにツリー64の識別タグであるタグID44を付加して送信する。

【0520】

ノード11～ノード14およびノード16の各ノードから、ノード15へフレームを送信する場合は、ツリー65を使用する。例えば、ノード16からノード15へフレームを送信する場合は、ノード16は、データフレームにツリー65の識別タグであるタグID45を付加して送信する。

【0521】

ノード11～ノード15の各ノードから、ノード16へフレームを送信する場合は、ツリー66を使用する。例えば、ノード14からノード16へフレームを送信する場合は、ノード14は、データフレームにツリー66の識別タグであるタグID46を付加して送信する。

【0522】

ノード11が、フレームをネットワーク内の全ノードにブロードキャスト送信する場合は、ツリー61を使用する。例えば、ノード11は、宛先がブロードキャストとなるデータフレームに対して、ツリー61の識別タグであるタグID41を付加して送信する。

【0523】

ノード12が、フレームをネットワーク内の全ノードにブロードキャスト送信

する場合は、ツリー62を使用する。例えば、ノード12は、宛先がブロードキャストとなるデータフレームに対して、ツリー62の識別タグであるタグID42を附加して送信する。

【0524】

ノード13が、フレームをネットワーク内の全ノードにブロードキャスト送信する場合は、ツリー63を使用する。例えば、ノード13は、宛先がブロードキャストとなるデータフレームに対して、ツリー63の識別タグであるタグID43を附加して送信する。

【0525】

ノード14が、フレームをネットワーク内の全ノードにブロードキャスト送信する場合は、ツリー64を使用する。例えば、ノード14は、宛先がブロードキャストとなるデータフレームに対して、ツリー64の識別タグであるタグID44を附加して送信する。

【0526】

ノード15が、フレームをネットワーク内の全ノードにブロードキャスト送信する場合は、ツリー65を使用する。例えば、ノード15は、宛先がブロードキャストとなるデータフレームに対して、ツリー65の識別タグであるタグID45を附加して送信する。

【0527】

ノード16が、フレームをネットワーク内の全ノードにブロードキャスト送信する場合は、ツリー66を使用する。例えば、ノード16は、宛先がブロードキャストとなるデータフレームに対して、ツリー66の識別タグであるタグID46を附加して送信する。

【0528】

以上に示す方法で送信時タグを附加してデータフレームを転送することで、前記データフレームを最低コスト経路で転送することができる。また、ルートノードが異なる複数のツリーを利用してフレームを転送するため、従来技術1および従来技術2に示したスパニングツリーのように、ルートノード付近にトラフィックが集中し、ルートノードから遠ざかるほどリンクの使用率が減るという現象が

発生せず、トラフィックの付加を分散できることがわかる。

#### 【0529】

次に、図23～図28において、ノードに障害が発生した場合の動作について、ノード12に障害が発生した場合を例にして説明する。なお、最初の状態では、ツリー61～ツリー66は、すでに構築されて安定しているとする。

#### 【0530】

ツリー61は、ノード12が障害によって停止すると、IEEE802.1wで規定されるラピッドスパニングツリー方式により、ノード13からノード11へのルートとして、リンク25、リンク27、リンク26、リンク23を経由するルートが選ばれ、ノード11へのフレーム及び、ノード11から各ノードへブロードキャストされるフレームの転送が継続される。

#### 【0531】

ツリー62は、ノード12が障害によって停止すると、ノード12がルートノードであるので、ツリーを再構成しなくてはならない。ノード12以外の別のノードがルートノードとなり、ノード12の回復前にツリー62の再構成が行われる。この再構成には、IEEE802.1Dでは数十秒、IEEE802.1wでは数秒間かかるが、そもそもツリー62は、各ノードからルート12へ向けて送信されるフレーム、および、ルート12から各ノードに向けてブロードキャスト送信されるフレームのためのツリーなので、再構成に時間がかかるても、ノード12以外の他のノード間で行われる通信には影響を与えない。

#### 【0532】

ツリー63は、ノード12が障害によって停止すると、IEEE802.1wで規定されるラピッドスパニングツリー方式により、ノード11からノード13へのルートとして、リンク23、リンク26、リンク27、リンク25を経由するルートが選ばれ、ノード13へのフレーム及び、ノード13から各ノードへブロードキャストされるフレームの転送が継続される。

#### 【0533】

ツリー64は、ノード12が障害によって停止すると、IEEE802.1wで規定されるラピッドスパニングツリー方式により、ツリーが再構成され、各ノ

ードからノード14へ送信されるフレーム及び、ノード14から各ノードへブロードキャストされるフレームの転送が継続される。

【0534】

ツリー65は、ノード12が障害によって停止すると、IEEE802.1wで規定されるラピッドスパニングツリー方式により、ツリーが再構成され、各ノードからノード15へのフレーム及び、ノード15から各ノードへブロードキャストされるフレームの転送が継続される。

【0535】

ツリー66は、ノード12が障害によって停止すると、IEEE802.1wで規定されるラピッドスパニングツリー方式により、ノード11からノード16へのルートとして、リンク23、リンク26、リンク27を経由するルートが選ばれ、ノード16へのフレーム及び、ノード16から各ノードへブロードキャストされるフレームの転送が継続される。

【0536】

次に、図29と図30を参照して、第1の実施の形態の図7におけるクライアントの一部が、デュアルホーミングによって、複数のノードと接続されている場合の、スパニングツリーの構成方法について述べる。

【0537】

図29において、クライアント97は、1つ以上のクライアントの集合であり、リンク87およびリンク88を通じて、ノード11およびノード16との間でフレームの送受信を行う機能を有する。

【0538】

リンク87は、クライアント97からノード15、及び、ノード15からクライアント97を結ぶ双方向リンクである。

【0539】

リンク88は、クライアント97からノード16、及び、ノード16からクライアント97を結ぶ双方向リンクである。

【0540】

図29におけるクライアント97のように、複数のノードと接続されたクライ

アント群が存在する場合は、前記クライアントを仮想的なノードとみなして、スパニングツリーの設定を行う。

【0541】

図30は、図29において、クライアント97を仮想ノード18と見立てた場合のネットワーク構成図である。

【0542】

スパニングツリー74は、ノード18をルートノードとするスパニングツリーである。ノード11～ノード16の各ノードから、ノード18へ送信されたフレームは、スパニングツリー74を利用してノード18、つまりクライアント97に到着する。また、クライアント97、すなわちノード18から送信されたブロードキャストフレームも、スパニングツリー74に沿って、ノード11～ノード16の各ノードにブロードキャストされる。

【0543】

なお、ノード18は仮想的なノードであり、ノード18の実際の動作は、ノード15もしくは16が代行して行う。ノード15とノード16のどちらがノード18の動作を代行するかは、設定インターフェース118による手動設定のほか、ノードIDの小さい方に自動設定、ノードIDの大きい方に自動設定するなどの方法で決定される。

【0544】

次に、図29を参照して、クライアント97が、リンク87およびリンク88を通じて、ノード15およびノード16とデュアルホーミング接続されている場合において、仮想ノードを設置することなく通信を行う方法について説明する。

【0545】

図29において、ノード15およびノード16は、設定インターフェースでの設定、もしくは、ラーニングによって、クライアント97が複数のノードに接続されていることを検知する。ノード15は、ノード16にクライアント97が接続されていることを検知する。ノード16は、ノード15にクライアント97が接続されていることを検知する。

【0546】

ノード15およびノード16は、相互で制御メッセージの交換を行い、クライアント97へのフレームを、ノード15もしくはノード16のどちらが転送するかを決定する。この転送ノードは、ノードIDの小さいノード、ノードIDの大きいノード、もしくは、設定で定められたノード等に決定される。

#### 【0547】

転送ノードが決定すると、クライアント97はノード16のみに接続されないとみなされ、フレーム転送が開始される。ノード11～ノード16は、クライアント97がノード16に接続されていると学習等で認識し、クライアント97宛のフレームを、ノード16をルートノードとするツリーの識別タグを附加して送信する。

#### 【0548】

ノード15およびノード16は、常にKeep Alive等により相互に状況を監視しあう。もし、ノード15がノード16の動作を確認できないときは、ノード15はクライアント97からのフレームを、ノード11～16に転送するようになる。すると、ノード11～ノード16は、クライアント97から送信されたフレームに、ノードIDとして15が付加されていることを学習し、クライアント97宛のフレームをノード15に向けて送信するようになる。

#### 【0549】

以上の動作により、クライアント97は、フレームを送受信することができる。以上の動作は、ノード15とノード16が入れ替わった場合においても、同様に適用可能である。

#### 【0550】

次に、図29を参照して、クライアント97が、リンク87およびリンク88を通じて、ノード15およびノード16とデュアルホーミング接続されている場合において、仮想ノードを設置することなく通信を行い、また、障害検出ノードがネットワーク内の全ノードに対して切替通知を送信することで、クライアント97の接続先が変更されることを高速に通知する方法について説明する。

#### 【0551】

図29において、ノード15およびノード16は、設定インターフェース118

での設定、もしくは、ラーニングによって、クライアント97が複数のノードに接続されていることを検知する。ノード15は、ノード16にクライアント97が接続されていることを検知する。ノード16は、ノード15にクライアント97が接続されていることを検知する。

#### 【0552】

ノード15およびノード16は、相互で制御メッセージの交換を行い、クライアント97へのフレームを、ノード15もしくはノード16のどちらが転送するかを決定する。この転送ノードは、ノードIDの小さいノード、ノードIDの大きいノード、もしくは、設定で定められたノード等に決定される。

#### 【0553】

転送ノードが決定すると、クライアント97はノード16のみに接続されないとみなされ、フレーム転送が開始される。ノード11～ノード16は、クライアント97がノード16に接続されていると学習等で認識し、クライアント97宛のフレームを、ノード16をルートノードとするツリーの識別タグを附加して送信する。

#### 【0554】

ノード15およびノード16は、常にKeep Alive等により相互に状況を監視しあう。もし、ノード15がノード16の動作を確認できないときは、ノード15はクライアント97からのフレームを、ノード11～16に転送するようになる。さらに、ノード15は、クライアント97宛のフレーム転送を、ノード16に代わってノード15が受け持つようになったことを、ネットワーク内の全ノードに通知する。

#### 【0555】

ノード11～ノード16は、前記通知を受け取り、クライアント97宛てのフレームに対してノード15宛のタグを挿入し、クライアント97宛のフレームをノード15に向けて送信するようになる。

#### 【0556】

以上の動作により、クライアント97は、フレームを送受信することができる。以上の動作は、ノード15とノード16が入れ替わった場合においても、同様

に適用可能である。

【0557】

次に、本実施の形態の効果について説明する。

【0558】

従来、宛先への最低コスト経路が必ずしも選択されるとは限らなかった。

【0559】

本実施の形態では、宛先がルートノードとなるツリーを用いてフレームを転送することにより、宛先への最低コスト経路を選択することができる。

【0560】

また従来、リンク利用率が低い一方、ルートノード付近に負荷が集中するという問題があった。

【0561】

本実施の形態では、ルートノードの異なるスパニングツリーを複数系統設定することにより、リンク利用率を高め、ルートノード付近に負荷を集中させずに分散させることが可能である。

【0562】

さらに従来、ルートノード障害時のツリー構築に時間がかかり、その間ネットワークが停止するという問題があった。

【0563】

本実施の形態では、宛先がルートノードとなるツリーを用いてフレームを転送することにより、ルートノードが宛先となるフレーム以外のフレームが、ルートノード障害の影響により長時間転送不能となることがないため、ルートノード障害によるネットワーク停止を回避可能である。

【0564】

またこの結果、輻輳発生の可能性を下げることが可能である。

【0565】

(第5の実施の形態)

以下、本発明の第5の実施の形態について、図面を参照して詳細に説明する。

【0566】

本発明の第5の実施の形態は、第1の実施の形態において、BPDUsのバージョンを識別し、低速なIEEE802.1Dすなわち従来技術1を用いている区間についてはコストを大きく設定し、高速なIEEE802.1wすなわち従来技術2を用いている区間についてはコストを小さく設定したスパニングツリーを生成する場合に対応する。

#### 【0567】

IEEE802.1Dを利用する区間は、障害時のルート切り替えが遅く、スパニングツリーの再構成にも時間がかかるため、この区間を通過するようなツリーを設定すると、障害発生時やルート変更に時間がかかり、輻輳が発生してフレームが欠落する問題があった。

#### 【0568】

本実施の形態においては、IEEE802.1D利用区間のコストを大きく設定することで、スパニングツリーがIEEE802.1D利用区間を通過して設定されるのを防ぎ、障害時の切替やルート変更を高速化し、輻輳の発生やフレームの欠落を防止する。

#### 【0569】

図31を参照すると、本発明の第5の実施の形態は、第1の実施の形態における図10において、コスト操作器11516が追加されている点において異なる。

#### 【0570】

ツリーコントローラ11514は、本発明の第1の実施の形態におけるツリーコントローラ11514の動作のほかに、受信したBPDUsのバージョンを判別し、もし設定されたバージョンより低いバージョンのBPDUsが到着した場合は、前記BPDUsを送信したノードが接続されているリンクのコストを、コスト操作器11516によって再設定し、ツリーテーブル11515に書き込む。なおこの動作は、ツリーセレクタ116よりコスト変更の通知を受信するごとに、1度のみ行う。

#### 【0571】

コスト操作器11516は、ツリーコントローラ11514より入力された値

に、あらかじめ設定された値を加算し、ツリーコントローラ11514に返答する。

【0572】

ツリーコントローラ11514は、BPDUs受信機11512よりをBPDUs受信通知を受けると、通知の内容に従ってツリーテーブル11515に値を設定していく。PDU受信通知には、受信したBPDUsのバージョンおよび受信ポートに関する情報も含まれ、前記情報もツリーテーブル11515において保持される。

【0573】

ツリーコントローラ11514は、ツリーセレクタ116より、コスト情報が通知されると、通知された情報に従ってテーブルにコスト値を設定する。このとき、あらかじめ設定されたバージョンより古いバージョンのBPDUsを受信したポートに対してコストを設定する場合は、ツリーセレクタ116より通知されたコストをコスト操作器11516に通知する。

【0574】

コスト操作器11516は、ツリーコントローラ11514より入力された値に、あらかじめ設定された値を加算し、ツリーコントローラ11514に返答する。

【0575】

ツリーコントローラ11514は、コスト操作器11516より返答されたコストを、該当ポートのコストとして、ツリーテーブル11515に通知する。

【0576】

ツリーコントローラ11514は、すべてのポートに対するコスト更新が完了した後で、スパニングツリーアルゴリズムにしたがってツリーを再構成する。

【0577】

次に、図32～図34を参照して、本実施の形態におけるスパニングツリー作成動作について述べる。

【0578】

図32～図34において、ノード12がIEEE802.1w非対応のノード

であり、IEEE802.1Dにのみ対応するとする。ノード12以外のノード、すなわち、ノード11、ノード13、ノード14、ノード15およびノード16は、IEEE802.1wに対応するとする。

#### 【0579】

ノード11、ノード15、ノード13の各ノードは、ノード12から送信されるBPDUフレーム内のバージョン情報もしくはプロトコルIDにより、ノード12が802.1Dのみに対応するノードであると認識する。

#### 【0580】

ノード11、ノード15、ノード13の各ノードは、リンク21、リンク22およびリンク24の各リンクのコストを、他のリンクのコストよりも十分大きく設定する。ここでは、リンク21、リンク22、リンク24のコストが10に設定され、他のリンク、すなわち、リンク23、リンク26、リンク27、リンク25のコストが1に設定されるとする。

#### 【0581】

図32は、前記コスト設定の状態において、ノード11もしくはノード14がルートノードとなる場合の、スパニングツリーの構成図である。

#### 【0582】

図33は、前記コスト設定の状態において、ノード15、ノード16もしくはノード14がルートノードとなる場合の、スパニングツリーの構成図である。

#### 【0583】

図34は、前記コスト設定の状態において、ノード13もしくはノード16がルートノードとなる場合の、スパニングツリーの構成図である。

#### 【0584】

図32～図34に示すように、本実施の形態では、障害回復に時間がかかるIEEE802.1D利用区間を避けるようにしてツリーを構成することが可能であり、障害時にネットワーク全体が受ける影響を小さくし、高速な障害回復ができる。

#### 【0585】

次に、本実施の形態の効果について説明する。

【0586】

従来、IEEE802.1Dを利用する区間は、障害時のルート切り替えが遅く、スパニングツリーの再構成にも時間がかかった。

【0587】

本実施の形態では、IEEE802.1D利用区間のコストを大きく設定することにより、スパニングツリーがIEEE802.1D利用区間を通過して設定されるのを防ぎ、障害時の切替やルート変更を高速化し、輻輳発生の可能性やフレームの欠落の可能性を低下させることができる。

【0588】

(第6の実施の形態)

以下、本発明の第6の実施の形態について、図面を参照して詳細に説明する。

【0589】

本発明の第6の実施の形態は、第1の実施の形態において、分別器がBPDUsのバージョンを識別し、ツリーセレクタを通じて、ツリーマネージャを低速なIEEE802.1D利用区間の数だけ作成することで、IEEE802.1D利用区間で障害が発生した場合に、前記区間を迂回する経路を高速に提供する場合に対応する。

【0590】

IEEE802.1Dを利用する区間は、障害時のルート切り替えが遅く、スパニングツリーの再構成にも時間がかかるため、この区間において障害が発生すると、障害発生時やルート変更に時間がかかり、輻輳が発生してフレームが欠落する問題があった。

【0591】

本実施の形態においては、IEEE802.1D利用区間の数だけツリーマネージャを作成し、相異なる1つのIEEE802.1D利用区間のコストを大きく設定したツリーをIEEE802.1D利用区間分作成し、障害等により前記区間を迂回する必要が生じた場合は、前記区間に對して大きなコストを振ったツリーを利用することで、迂回操作を高速に行い、輻輳の発生やフレームの欠落を防止できる。

## 【0592】

図35を参照すると、本発明の第6の実施の形態は、本発明の第1の実施の形態における図8において、ツリーマネージャ1151がIEEE802.1D利用区間の数だけ存在する点において異なる。

## 【0593】

ツリーマネージャ1151は、本発明の第1の実施の形態における図8に示すツリーマネージャ1151の機能のほか、受信したBPDUフレームが、バージョンフィールドもしくはその他の手段によりIEEE802.1Dに準拠したBPDUフレームであると確認できた場合は、ツリーセレクタ116に対して、前記BPDUフレームに対して802.1Dフレーム受信通知を送信する。802.1Dフレーム受信通知には、802.1D準拠のBPDUフレームを送信したノードのノードIDが記入される。

## 【0594】

ツリーマネージャ1152および、ツリーマネージャ1153は、ツリーマネージャ1151と同様のツリーマネージャである。以後、ツリーマネージャ1151～ツリーマネージャ1153を代表してツリーマネージャ1151を用いて記述を行うが、ツリーマネージャ1151についての記述は、特に断りのない限りツリーマネージャ1152～ツリーマネージャ1153においても同様に適用可能である。

## 【0595】

ツリーマネージャはツリーセレクタ116によってIEEE802.1D利用区間の数だけ作成される。したがって、ツリーマネージャは、1個以上無限大まで増える可能性があるが、図35においては前記ツリーマネージャを、ツリーマネージャ1151～1153としてまとめて表現する。

## 【0596】

ツリーセレクタ116は、本発明の第1の実施の形態における図8に示すツリーセレクタ116の機能のほか、ツリーマネージャ1151～ツリーマネージャ1153の何れかのツリーマネージャより802.1Dフレーム受信通知を受けた場合は、新規のツリーマネージャを生成する機能、802.11D利用ノード

をネットワーク内の他のノードに通知する機能、および、他のノードより送信された、802.1D利用ノードの通知に基づき、ツリーマネージャを生成する機能を有する。

【0597】

ツリーセレクタ116は、前記機能のほか、802.1D利用ノードがバージョンアップ等の何らかの原因により802.1wが利用可能になったことを検知し、ツリーマネージャを削除する機能、そして、前記削除に関する情報を、ネットワーク内の他のノードに伝達する機能、および、他のノードより通知された前記削除情報に基づき、ツリーマネージャを削除する機能を有する。

【0598】

図36は、ノード11をルートノードとし、IEEE802.1wの通常の手順によって作成される、ツリー67の構成図である。

【0599】

図37は、ノード11をルートノードとし、ツリー21のコストを大きくして作成される、ツリー68の構成図である。このツリーは、リンク21に障害が発生した際にも利用される。

【0600】

図38は、ノード11をルートノードとし、ツリー22のコストを大きくして作成される、ツリー69の構成図である。このツリーは、リンク22に障害が発生した際にも利用される。

【0601】

図39は、ノード11をルートノードとし、ツリー24のコストを大きくして作成される、ツリー70の構成図である。このツリーは、リンク24に障害が発生した際にも利用される。

【0602】

次に、図36～図39を参照して、図36～図39におけるノード12がIEEE802.1w未対応のノードであり、リンク21とリンク22とリンク24において、IEEE802.1Dが利用されている場合の動作について述べる。なお、ルートノードは、ノード11とする。

## 【0603】

まず、IEEE802.1wに従った通常の手順によって、図36に示すスパンニングツリー67が形成される。このとき、ノード12は802.1w未対応のノードであるため、ノード12からは、IEEE802.1DのプロトコルIDが付加されたBPDUフレームが送信される。

## 【0604】

ノード11は、ノード12よりIEEE802.1DのプロトコルIDが付加されたBPDUを受信すると、新規のツリーマネージャを生成して、前記ツリーマネージャに対して、リンクIDおよびノードID等から算出する固有のタグIDを割り当て、GVRPフレームもしくはその他のフレームによって、全ノードに対して新規グループの作成をブロードキャスト通知する。ここでは新規のタグIDとして、タグID48が割り当てられたとする。このとき、リンク21のコストは大きく設定する。

## 【0605】

ノード12～ノード16は、ノード11より送信された新規グループ作成通知を受信転送し、ツリーマネージャを生成してBPDUの交換を開始する。新規に作成されたツリーマネージャの間で交換されるBPDUには、タグID48のタグが付加される。ここで作成されるスパンニングツリーを、ツリー68とする。

## 【0606】

ノード13は、ノード12よりIEEE802.1DのプロトコルIDが付加されたBPDUを受信すると、新規のツリーマネージャを生成して、前記ツリーマネージャに対して、リンクIDおよびノードID等から算出する固有のタグIDを割り当て、GVRPフレームもしくはその他のフレームによって、全ノードに対して新規グループの作成をブロードキャスト通知する。ここでは新規のタグIDとして、タグID49が割り当てられたとする。このとき、リンク22のコストは大きく設定する。

## 【0607】

ノード11～ノード12およびノード14～ノード16は、ノード13より送信された新規グループ作成通知を受信転送し、ツリーマネージャを生成してBP

D Uの交換を開始する。新規に作成されたツリーマネージャの間で交換されるB P D Uには、タグ I D 4 9のタグが付加される。ここで作成されるスパニングツリーを、ツリー 6 9とする。

#### 【0608】

ノード15は、ノード12よりIEEE802.1DのプロトコルI Dが付加されたB P D Uを受信し、新規のツリーマネージャを生成して、前記ツリーマネージャに対して、リンク I Dおよびノード I D等から算出する固有のタグ I Dを割り当て、G V R P フレームもしくはその他のフレームによって、全ノードに対して新規グループの作成をブロードキャスト通知する。ここでは新規のタグ I Dとして、タグ I D 5 0が割り当てられたとする。このとき、リンク 2 4のコストは大きく設定する。

#### 【0609】

ノード11～ノード14およびノード16は、ノード15より送信された新規グループ作成通知を受信転送し、ツリーマネージャを生成してB P D Uの交換を開始する。新規に作成されたツリーマネージャの間で交換されるB P D Uには、タグ I D 5 0のタグが付加される。ここで作成されるスパニングツリーを、ツリー 7 0とする。

#### 【0610】

平常時においては、各ノード間の通信はツリー 6 7を用いて行われ、ツリー 6 8～ツリー 7 0は利用されない。

#### 【0611】

ここで、もしリンク 2 1に障害が発生したとすると、ノード11は、リンク 2 1の障害を検知して、直ちに転送に使用するツリーを、ツリー 6 7からツリー 6 8に切り替える。また、利用タググループ変更通知を全ノードに対してブロードキャスト送信し、転送に使用するタグをタグ I D 4 8に切り替えるように通知する。

#### 【0612】

ノード12～ノード16の各ノードは、ノード11より送信された利用タググループ変更通知を受信し、自ノードから送信するフレームに対してタグ I D 4 8

のタグを挿入し、転送に使用するツリーを、ツリー67からツリー68に切り替える。

【0613】

なお、リンク21に障害が発生した場合の、ツリー68への切替は、ノード11ではなくノード12が行ってもかまわない。もしノード12がリンク21の障害を検出した場合は、直ちに転送に使用するツリーを、ツリー67からツリー68に切り替える。また、利用タググループ変更通知を全ノードに対してブロードキャスト送信し、転送に使用するタグをタグID48に切り替えるように通知する。以降の動作は、ノード11が障害を検出した場合と同様である。

【0614】

ツリー67は、リンク21の障害によって再構成されるが、リンク21では802.1Dに従った再構成が行われるために、再構成が完了するまでに時間がかかる可能性がある。

【0615】

本実施の形態では、リンク21の障害時に、転送に利用されるツリーを、ツリー67からツリー68にただちに切り替えることで、ツリー67の再構成を待たずに、フレームの転送を継続することができる。

【0616】

以上、リンク21に障害が発生した場合の動作について述べたが、前記動作はリンク22もしくはリンク24に障害が発生した場合においても、同様に適用可能である。

【0617】

次に、本実施の形態の効果について説明する。

【0618】

従来、IEEE802.1Dを利用する区間は、障害時のルート切り替えが遅く、スパンニングツリーの再構成にも時間がかかった。

【0619】

本実施の形態では、IEEE802.1D利用区間の数だけツリーマネージャを作成し、相異なる1つのIEEE802.1D利用区間のコストを大きく設定

したツリーを IEEE 802. 1 D 利用区間分作成し、障害等により前記区間を迂回する必要が生じた場合は、前記区間に對して大きなコストを振ったツリーを利用するよう切り替えることで、迂回操作を高速に行い、輻輳発生の可能性やフレームの欠落の可能性を低下させることができる。

#### 【0620】

(第7の実施の形態)

次いで、本発明の第7の実施の形態について、図面を参照して詳細に説明する。

#### 【0621】

本発明の第7の実施の形態は、第1の実施の形態において、障害検出器を用いて定期的に短い間隔で障害検出用フレームを送信し、前記障害検出用フレームが到着しなくなったことにより、障害検出を行い、障害情報をリソースモニタおよびツリーセレクタを通じて、ツリーマネージャに通知する点が異なる。

#### 【0622】

IEEE 802. 1 D および IEEE 802. 1 w では、定期的に送信される HELLO フレームの未着により、障害を検出していた。しかしながら、HELLO フレームの送信間隔は長いため、障害を検出するまでに長い時間がかかった。

#### 【0623】

本実施の形態では、障害検出器から定期的に短い間隔で障害検出用フレームを送信し、前記障害検出用フレームの一定個数もしくは一定時間以上の未着により、高速な障害検出を行う。

#### 【0624】

図40を参照すると、本発明の第7の実施の形態は、第1の実施の形態におけるおける図4において、障害検出用フレームを送受信して障害を検出する、障害検出器120が追加されている点において異なる。

#### 【0625】

障害検出器120は、障害検出用フレームをフレーム転送器111を通じて隣接ノードに対して定期的に送信し、また、隣接ノードより送信された障害検出用

フレームを、フレーム転送器111より受信し、隣接ノードから送られる障害検出用フレームが一定時間以上受信できない場合、および、一定個数以上の障害検出用フレームの未着を検出した場合に、リソースモニタ119に対して障害検出通知を送信する。

## 【0626】

リソースモニタ119は、第1の実施の形態におけるリソースモニタ119の機能のほか、障害検出器120より障害通知を受信し、前記障害通知をツリーセレクタ116に転送する機能を有する。

## 【0627】

ツリーセレクタ116は、第1の実施の形態におけるツリーセレクタ116の機能のほか、リソースモニタ119より障害通知を受信し、これをツリーマネージャ1151および1152に転送する機能を有する。

## 【0628】

ツリーマネージャ1151および1152は、本発明の第1の実施の形態におけるツリーマネージャ1151および1152の機能のほか、ツリーセレクタ116より障害通知を受信し、IEEE802.1wもしくはIEEE802.1Dに従い、スパニングツリーの再構成を行う。

## 【0629】

次に、図40を参照して、本実施の形態においてノード11がリンク21の障害を検知する場合の動作例について述べる。

## 【0630】

障害検出器120は、定期的に、フレーム転送器111およびリンク21もしくはリンク24を経由して、隣接ノードであるノード12およびノード15に障害検出用フレームを送信する。

## 【0631】

障害検出器120は、また、隣接しているノード12およびノード15より、リンク21もしくはリンク24と、フレーム転送器111を経由して送信された障害検出用フレームを受信する。このとき、障害検出器120は、前記障害検出用フレームが到着したポートのIDも、識別することができる。

## 【0632】

障害検出器120は、障害検出用フレームが到着すると、前記障害検出用フレームが到着したポートのタイマを動作させ、一定時間が経過すると、通知を発信するように設定する。

## 【0633】

障害検出器120は、障害検出用フレームを受信すると、ポートごとに設置されたタイマをリセットする。例えば、リンク21より障害検出用フレームを受信した場合は、リンク21が接続されているポートのタイマがリセットされる。また、リンク24より障害検出用フレームを受信した場合は、リンク24が接続されているポートのタイマがリセットされる。

## 【0634】

ここで、障害検出器120において、もしリンク障害等により、障害検出用フレームが一定時間以上到着しない場合は、タイマがリセットされないためタイムアウトが起こる。障害検出器120は、タイムアウトが発生すると、何らかの障害が発生していると認識し、リソースモニタ119およびツリーセレクタ11.6を通じて、ツリーマネージャ1151およびツリーマネージャ1152に将棋ア発生を通知する。

## 【0635】

通知を受けたツリーマネージャ1151およびツリーマネージャ1152は、障害発生ポートが使用不可能なものとして、直ちにスパニングツリーを再構成し、障害回避を行う。

## 【0636】

次に、本実施の形態の効果について説明する。

## 【0637】

従来、スパニングツリープロトコルで利用されているHELLOフレームの送信間隔が長かったため、高速な障害検出ができなかった。

## 【0638】

本実施の形態では、短い間隔で障害検出用フレームを送受信する障害検出器を追加することにより、HELLOフレームよりも高速な障害検出が可能である。

## 【0639】

また、この結果、輻輳発生およびフレーム欠落の可能性を下げることが可能である。

## 【0640】

## (実施の形態8)

以下、本発明の第8の実施の形態について、図面を参照して詳細に説明する。

## 【0641】

本発明の第8の実施の形態は、第1の実施の形態において、宛先ノード別に利用するタグおよびスパンニングツリーを切り替え、さらに、宛先のノードをルートノードに設定することができる構成となっている。

## 【0642】

IEEE802.1DおよびIEEE802.1wが動作しているネットワークにおいてフレームを送信した場合、宛先への最低コスト経路が必ずしも選択されるとは限らず、使用されないリンクが出現するほか、ルートノードに負荷が集中する、ルートノード障害時に長時間ネットワークが停止するなどの問題があった。

## 【0643】

本実施の形態においては、宛先がルートノードとなるツリーを用いてフレームを転送することで、宛先への最低コストによるフレーム伝送のほか、リンク利用率の向上、さらにはルートノード障害への耐性強化を実現する。

## 【0644】

図41を参照すると、本発明の第8の実施の形態は、第1の実施の形態における図8において、フレーム転送器111がフレーム転送器111'に、フォワーディングテーブル114がフォワーディングテーブル114'になり、さらにツリーマネージャ1151が、動作を変更した上でネットワークに存在するノードの数だけ設置され、ツリーマネージャ1151'～1153'になっている点において異なる。

## 【0645】

図41においては、本実施の形態を図23に示すノード11に適用した場合を

示している。

【0646】

フレーム転送器111 $\gamma$ は、リンク21もしくはリンク23、およびタグ挿入器112より受信したフレームを、フォワーディングテーブル114 $\gamma$ の出力ポート1142の記述に従い、リンク21もしくはリンク23、および、タグ削除器113もしくはツリーセレクタ116に転送する。この際、もし出力ポート1142の記述が初期値である場合は、前記受信フレームを廃棄する。

【0647】

フォワーディングテーブル114 $\gamma$ は、フォワーディングテーブル114と同様のフォワーディングテーブルである。

【0648】

ツリーマネージャ1151 $\gamma$ は、ツリーマネージャ1151と構成は同じであるが、その機能と動作において異なっている。以後、ツリーマネージャ1151 $\gamma$ ～ツリーマネージャ1153 $\gamma$ を代表して、ツリーマネージャ1151 $\gamma$ を用いて記述を行うが、ツリーマネージャ1151 $\gamma$ についての記述は、特に断りのない限りツリーマネージャ1152 $\gamma$ ～ツリーマネージャ1153 $\gamma$ においても同様に適用可能である。

【0649】

図42は、フォワーディングタグをキーとして出力ポートを決定する、フォワーディングテーブル114 $\gamma$ の構成例である。

【0650】

タグフィールド1141は、フォワーディングテーブル114内のタグフィールドと同様のタグフィールドである。

【0651】

出力ポート1142は、フォワーディングテーブル114内の出力ポートと同様の出力ポートであり、ユニキャストフレームの転送時にフレーム転送器111 $\gamma$ によって参照される。

【0652】

図42において、出力ポート1142の「END」は、エッジノードにおいて

宛先ノードIDに対する出力ポートエントリが自ノードであることを示す識別子であり、例えば自ノードIDと等しいノード識別子エントリに対して出力ポートフィールドに記述される。

#### 【0653】

図43は、ツリーマネージャ1151γの構成を示す、ブロック図である。

#### 【0654】

図23に示すようにスパニングツリーが構成されている場合、各ノードのポートは、そのスパニングツリーの構成に応じて、図54のようにルートポート（Root Port）、ディジグネイテッドポート（Designated Port）又はオルタネートポート（Alternate Port）として決定される。図54においては、ルートポートを（R）、ディジグネイテッドポートを（D）、オルタネートポートを（ALT）として示している。もちろん、上記ポートの種別は、スパニングツリーの構成（ルートノードの位置）によって変化する。

#### 【0655】

ツリーマネージャ1151γが送受信するBPDUsに、タググループ41のタグが付加されている場合、ツリーコントローラ11514γは、IEEE802.1wもしくはIEEE802.1Dプロトコルによりルートポート（Root Port）を決定する。本実施の形態においてはフォワーディングテーブル114γの、タググループ41のエントリに、出力ポート1142として決定したルートポートを設定する。この場合、最低6個のBPDUsが送受信される。

#### 【0656】

ここで、もしルートポートが存在しない場合（図54のノード11）には、出力ポート1142の欄は、自ノード宛ポートに設定し、フレーム転送器に入力されたタググループ41のタグつきフレームを、タグ削除器113に転送する。

#### 【0657】

次に、図23および図44を用いて、ユニキャストフレーム転送動作について、ノード13からノード11へのユニキャストフレーム転送を例に説明する。

図44は、スパニングツリー61における各ノードのポート設定およびフォワー

ディングテーブルの設定の状況示す表である。なお、この図44は、実際のテーブル内容を示すものではなく、あくまで設定状況を示すものである。

#### 【0658】

図23に示す状態において、タググループ41のツリーであるツリー61は、すでに構築されて安定しており、ノード11～ノード16の各ノードにおけるルートポートは、図44におけるルートポート6102に示すように決定されており、この結果より、各ノードのフォワーディングテーブル114における出力ポートは、図44における出力ポート1142に示すように決定されているものとする。

#### 【0659】

ここで、ツリー61は、ノード11をルートノードとするツリーを表している。また、タグ41は、ツリー61を示すタグのID(値)を表している。すなわち、タグ41をフレームに付加するということは、ツリー61を用いてフレーム転送することを意味する。このように、ツリーの番号とノードの番号とタグの番号とのそれは、一の位の数字がそれぞれ対応することを示している。例えばツリー62は、ノード12をルートとするツリーを示しており、タグ42は、ツリー62を示すタグのID(値)を表している、この点は、以下の説明においても同様である。

#### 【0660】

また、ここでは一例として上述のように、ツリー、ノード、タグにそれぞれ異なる番号を付した場合について説明をしたが、これ以外に、それぞれ対応するツリー、ノード、タグに同一の番号(ID)を付して、それぞれの対応関係を単純明確化して管理することも可能である。

#### 【0661】

まず、ノード13がノード11宛のユニキャストフレームに、タググループ41のタグを付加してフレームを送信する。この際、ノード13におけるタググループ41のフレームの出力先ポートは、ツリー61のルートポートであるリンク22側のポートが指定されている。したがって、前記フレームはリンク22側に出力される。

## 【0662】

ノード12は、リンク22よりフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、出力ポートとしてリンク21側のポートを得る。そして、リンク21側に前記受信フレームを出力する。

## 【0663】

ノード11は、リンク21よりフレームを受信すると、自ノード宛であることを確認し、タグ削除器113にフレームを転送する。

## 【0664】

以上の動作により、タググループ41のタグおよびスパニングツリー61を用いて、最低コスト経路によって、ノード13からノード11へのユニキャストフレームを転送することが可能となる。

## 【0665】

次に、上述した第8の実施の形態の効果について説明する。

## 【0666】

従来、宛先への最低コスト経路が必ずしも選択されることは限らなかったが、本実施の形態では、宛先がルートノードとなるツリーを用いてフレームを転送することにより、宛先への最低コスト経路を選択することができる。

## 【0667】

また従来、リンク利用率が低い一方、ルートノード付近に負荷が集中するという問題があったが、本実施の形態では、ルートノードの異なるスパニングツリーを複数系統設定することにより、リンク利用率を高め、ルートノード付近に負荷を集中させずに分散させることができる。

## 【0668】

さらに従来、ルートノード障害時のツリー構築に時間がかかり、その間ネットワークが停止するという問題があったが、本実施の形態では、宛先がルートノードとなるツリーを用いてフレームを転送することにより、ルートノードが宛先となるフレーム以外のフレームが、ルートノード障害の影響により長時間転送不能となることがないため、ルートノード障害によるネットワーク停止を回避可能である。また、この結果、輻輳発生の可能性を下げることが可能である。

## (実施の形態9)

以下、本発明の第9の実施の形態について、図面を参照して詳細に説明する。

## 【0669】

本発明の第9の実施の形態は、第8の実施の形態において、フォワーディングテーブルに、通常のユニキャスト用出力ポートのほかに、複数のブロードキャスト用出力ポートを記載し、ブロードキャストフレームを転送することができる構成となっている。

## 【0670】

本実施の形態においては、ブロードキャストフレームの送信元ノードがルートノードとなるツリーを用いてブロードキャストフレームを転送することで、ブロードキャストフレームを各ノードに最短経路で送信することができ、高速な転送が可能になる。

## 【0671】

図4.5を参照すると、本発明の第9の実施の形態は、第8の実施の形態における図4.1において、フレーム転送器111がフレーム転送器111 $\beta$ に、フォワーディングテーブル114がフォワーディングテーブル114 $\beta$ に、ツリーマネージャ1151～1153がツリーマネージャ1151 $\beta$ ～1153 $\beta$ になっている点において異なる。

## 【0672】

図4.5においては、本実施の形態を図2.3に示すノード11に適用した場合を示している。

## 【0673】

フレーム転送器111 $\beta$ は、リンク21もしくはリンク24、およびタグ挿入器112より受信したフレームを、フォワーディングテーブル114 $\beta$ の記述に従い、リンク21もしくはリンク24、および、タグ削除器113もしくはツリーセレクタ116に転送する。

## 【0674】

この際、もし入力されたフレームがユニキャストフレームである場合は、前記受信フレームをフォワーディングテーブル114 $\beta$ の出力ポート1142に記述

されたポートに転送する。

【0675】

もし、入力されたフレームがブロードキャストフレームである場合は、フォワーディングテーブル114βのブロードキャスト出力ポート1144に記述された複数のポートおよびタグ削除器に、前記受信フレームをコピーして転送する。さらにもし、ブロードキャスト出力ポートに初期値が設定されていた場合は、前記受信フレームをタグ削除器のみに転送する。

【0676】

ブロードキャストフレームとユニキャストフレームの判別は、送信先MACアドレス3201、もしくは拡張タグのプライオリティ5003、もしくは拡張タグ情報フィールド5004によって行う。

【0677】

フォワーディングテーブル114βは、フォワーディングテーブル114に、ブロードキャスト出力ポート1144欄が付加されたフォワーディングテーブルである。ブロードキャスト出力ポート1144欄は、対応するタグIDが示すノードから送信されたブロードキャスト信号を受信したときに、転送先となるポートを示している。本実施形態におけるフォワーディングテーブルの一例を図46に示す。図46では、転送先ポートを示すものとして、リンク名を用いている。図46は、図23～28に記載の物理トポジからなるネットワークにおけるノード11のフォワーディングテーブル114βを示している。上述の通り、本発明では、宛先ノード毎に、フレーム転送に用いるツリーが異なっている。例えば、ノード11宛のフレームは、図23のツリーを用い、ノード12宛のフレームは図24のツリーを用いてフレーム転送を行う。

【0678】

ここで、タグ42が付加されたフレームのブロードキャストの場合を例として以下に図46の見方を説明する。タグ42が付加されているということは、ノード12から発信されたブロードキャストフレームであることを示している。そして、ノード12から発信されたブロードキャストフレームは、本発明では、図24のツリーを用いて転送される。従って、ノード11では、このブロードキャス

トフレームを受信すると、リンク23側に転送する必要がある。このような考え方に基づいて、図46は作成される。

【0679】

ツリーマネージャ1151 $\beta$ は、ツリーマネージャ1151と構成は同じであるが、その機能と動作において異なっている。以後、ツリーマネージャ1151 $\beta$ ～ツリーマネージャ1153 $\beta$ を代表して、ツリーマネージャ1151 $\beta$ を用いて記述を行うが、ツリーマネージャ1151 $\beta$ についての記述は、特に断りのない限りツリーマネージャ1152 $\beta$ ～ツリーマネージャ1153 $\beta$ においても同様に適用可能である。

【0680】

図46は、タグをキーとして出力ポートを決定する、フォワーディングテーブル114 $\beta$ の構成例である。

【0681】

タグフィールド1141は、フォワーディングテーブル114内のタグフィールドと同様のタグフィールドである。

【0682】

出力ポート1142は、フォワーディングテーブル114内の出力ポートと同様の出力ポートであり、ユニキャストフレームの転送時にフレーム転送器111 $\beta$ によって参照される。

【0683】

ブロードキャスト出力ポート1144は、ブロードキャストフレームの転送時に、フレーム転送器111 $\beta$ によって参照される出力ポートである。この欄には複数のポートが記載され、もし2つ以上のポートが記載された場合は、記載されたポートの数だけフレームをコピーしてから転送する。もし、フレーム転送器111 $\beta$ によって参照されたときに、この欄に設定された値が初期値のままである場合は、フレーム転送器111 $\beta$ は、フレームをタグ削除器のみに転送する。

【0684】

なお、図46中の（END）とあるのは、ノードがエッジノードである場合に識別子「END」が記述されることを示している。

## 【0685】

図47は、ツリーマネージャ1151 $\beta$ の構成を示す、ブロック図である。

## 【0686】

ツリーマネージャ1151 $\beta$ が送受信するBPDUsに、タググループ41のタグが付加されている場合、ツリーコントローラ11514 $\beta$ は、IEEE802.1wもしくはIEEE802.1Dプロトコルによりルートポート（Root Port）およびディジグネイティッドポート（Designated Port）を決定する。本実施の形態では、フォワーディングテーブル114 $\beta$ の、タググループ41のエントリに、出力ポート1142として決定したルートポートを、ブロードキャスト出力ポート1144として、1つまたは複数の決定したディジグネイティッドポートを、それぞれ設定する。

## 【0687】

ここで、もしルートポートが存在しない場合には、出力ポート1142の欄は、自ノード宛ポートに設定し、フレーム転送器に入力されたタググループ41のタグつきフレームを、タグ削除器113に転送する。

## 【0688】

また、もしディジグネイティッドポートが1つも存在しない場合には、ブロードキャスト出力ポート1144の欄は、テーブルの初期値に設定する。

## 【0689】

次に、図23および図48を用いて、ブロードキャストフレーム転送動作について説明する。

## 【0690】

図48は、スパニングツリー61における各ノードのポート設定およびフォワーディングテーブルの設定を示す表である。

## 【0691】

最初の状態では、タググループ41のツリーであるツリー61は、すでに構築されて安定しており、ノード11～ノード16の各ノードにおけるルートポートおよびディジグネイティッドポートは、図48におけるルートポート6102およびディジグネイティッドポート6104に示すように決定されており、この結果よ

り、各ノードのフォワーディングテーブル114βにおける出力ポートおよび予備出力ポートは、図48における出力ポート1142およびブロードキャスト出力ポート1144に示すように決定されているものとする。

#### 【0692】

まず、ノード11がブロードキャストフレームに、タググループ41のタグを付加して送信する。この際、ノード11におけるタググループ41のブロードキャストフレームの出力先ポートは、ツリー61のデイジグネイティッドポートであるリンク21側およびリンク23側のポートが指定されている。したがって、前記フレームはリンク21側およびリンク23側、さらにタグ削除器宛に、コピーされてから出力される。

#### 【0693】

ノード12は、リンク21よりブロードキャストフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、ブロードキャスト出力ポートとしてリンク22側のポートを得る。そして、リンク22側およびタグ削除器に、前記受信ブロードキャストフレームを出力する。

#### 【0694】

ノード13は、リンク22よりブロードキャストフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、ブロードキャスト出力ポートとして初期値を得る。そして、タグ削除器に前記受信フレームを出力する。

#### 【0695】

ノード14は、リンク23よりブロードキャストフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、ブロードキャスト出力ポートとしてリンク26側のポートを得る。そして、リンク26側およびタグ削除器に、前記受信フレームを出力する。

#### 【0696】

ノード15は、リンク26よりブロードキャストフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、ブロードキャスト出力ポートとしてリンク27側のポートを得る。そして、リンク27側および

タグ削除器に、前記受信フレームを出力する。

【0697】

ノード16は、リンク27よりブロードキャストフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、ブロードキャスト出力ポートとして初期値を得る。そして、タグ削除器に前記受信フレームを出力する。

【0698】

以上の動作により、ノード11から出力されたブロードキャストフレームは、最低コスト経路によって、ネットワーク上の各ノードに転送される。

【0699】

次に、上述した第9の実施の形態の効果について説明する。

【0700】

従来、ブロードキャスト時に、宛先への最低コスト経路が必ずしも選択されるとは限らなかったが、本実施の形態では、送信元ノードがルートノードとなるツリーを用いてブロードキャストフレームを転送することにより、全ノードへの最低コスト経路を選択してブロードキャストフレームを転送することができる。

【0701】

また従来、リンク利用率が低い一方、ルートノード付近に負荷が集中するという問題があったが、本実施の形態では、ルートノードの異なるスパニングツリーを複数系統設定することにより、リンク利用率を高め、ルートノード付近に負荷を集中させずに分散させることができる。

【0702】

さらに従来、ルートノード障害時のツリー構築に時間がかかり、その間ネットワークが停止するという問題があったが、本実施の形態では、送信元ノードがルートノードとなるツリーを用いてブロードキャストフレームを転送することにより、ルートノードが送信元ノードとなるフレーム以外のブロードキャストフレームが、ルートノード障害の影響により長時間転送不能となることがないため、ルートノード障害によるネットワーク停止を回避可能である。また、この結果、輻輳発生の可能性を下げることが可能である。

【0703】

(実施の形態10)

以下、本発明の第10の実施の形態について、図面を参照して詳細に説明する

【0704】

本発明の第10の実施の形態は、第8の実施の形態において、フォワーディングテーブルに出力ポートを2つ記載し、1つの出力ポートが障害等で利用できない場合に、もう一方のポートを利用できるようにし、さらに、第7の実施の形態における障害検出器を利用し、高速な障害検出を行うことができる構成となっている。

【0705】

本実施の形態においては、宛先がルートノードとなるツリーを用いてユニキャストフレームを転送し、さらにスパニングツリーによって決定される代替出力ポートをあらかじめフォワーディングテーブルに登録することで、高速な障害回復を行う。

【0706】

図49を参照すると、本発明の第8の実施の形態は、第4の実施の形態における図41において、フレーム転送器111がフレーム転送器111 $\alpha$ に、フォワーディングテーブル114がフォワーディングテーブル114 $\alpha$ に、ツリーマネージャ1151～1153がツリーマネージャ1151 $\alpha$ ～1153 $\alpha$ になり、さらに、第7の実施の形態に記した障害検出器120が追加されている点において異なる。

【0707】

図49においては、本実施の形態を図23に示すノード11に適用した場合を示している。

【0708】

フレーム転送器111 $\alpha$ は、リンク21もしくはリンク23、およびタグ挿入器112より受信したフレームを、フォワーディングテーブル114 $\alpha$ の記述に従い、リンク21もしくはリンク23、および、タグ削除器113もしくはツリ

ーセレクタ116に転送する。

【0709】

この際、もしリソースモニタ119より、フォワーディングテーブル114 $\alpha$ の出力ポート1142に記述されたポートに障害があることを検知している場合は、前記受信フレームを予備出力ポート1143に記述されたポートに転送する。もし、出力ポート1142に記述されたポートの障害を検知しているが、予備出力ポートの記述が初期値（又は未設定）である場合は、前記受信フレームを廃棄する。

【0710】

フォワーディングテーブル114 $\alpha$ は、フォワーディングテーブル114に、予備出力ポート1143欄が付加されたフォワーディングテーブルである。

【0711】

ツリーマネージャ1151 $\alpha$ は、ツリーマネージャ1151と構成は同じであるが、その機能と動作において異なっている。以後、ツリーマネージャ1151 $\alpha$ ～ツリーマネージャ1153 $\alpha$ を代表して、ツリーマネージャ1151 $\alpha$ を用いて記述を行うが、ツリーマネージャ1151 $\alpha$ についての記述は、特に断りのない限りツリーマネージャ1152 $\alpha$ ～ツリーマネージャ1153 $\alpha$ においても同様に適用可能である。

【0712】

図50は、タグをキーとして出力ポートを決定する、図23におけるノード12のフォワーディングテーブル114 $\alpha$ の構成例である。

【0713】

タグフィールド1141は、フォワーディングテーブル114内のタグフィールドと同様のタグフィールドである。

【0714】

出力ポート1142は、フォワーディングテーブル114内の出力ポートと同様の出力ポートである。

【0715】

予備出力ポート1143は、出力ポート1142に記載されたポートが利用で

きなくなった場合に使用する出力先ポートを記載するためのフィールドである。出力ポート1142に記載されたポートが利用不能であることをフレーム転送器111 $\alpha$ が検知した場合、フレーム転送器111は当該エントリの予備出力ポート1143に記載されたポートにフレームを転送する。

## 【0716】

図51は、ツリーマネージャ1151 $\alpha$ の構成を示す、ブロック図である。

## 【0717】

ツリーマネージャ1151 $\alpha$ が送受信するBPDUsに、タググループ41のタグが付加されている場合、ツリーコントローラ11514 $\alpha$ は、IEEE 802.1wプロトコルによりルートポート（Root Port）およびオルタネートポート（Alternate Port）を決定する。本実施の形態では、フォワーディングテーブル114 $\alpha$ の、タググループ41のエントリに、出力ポート1142として決定したルートポートを、予備出力ポート1143として決定したオルタネートポートを、それぞれ設定する。

## 【0718】

ここで、もしルートポートが存在しない場合には、出力ポート1142の欄は、自ノード宛先ポートに設定し、フレーム転送器に入力されたタググループ41のタグつきフレームを、タグ削除器113に転送する。

## 【0719】

また、もしオルタネートポートが存在しない場合には、予備出力ポート1143の欄は、テーブルの初期値に設定する。

## 【0720】

次に、図23および図52を用いて、リンクに障害が発生した場合のユニキャストフレーム転送動作について、リンク21に障害が発生した場合を例にして説明する。

図52は、スパンニングツリー61における各ノードのポート設定およびフォワーディングテーブルの設定を示す表である。

## 【0721】

最初の状態では、タググループ41のツリーであるツリー61は、すでに構築

されて安定しており、ノード11～ノード16の各ノードにおけるルートポートおよびオルタネートポートは、図52におけるルートポート6102およびオルタネートポート6103に示すように決定されており、この結果より、各ノードのフォワーディングテーブル114 $\alpha$ における出力ポートおよび予備出力ポートは、図52における出力ポート1142および予備出力ポート1143に示すように決定されているものとする。

#### 【0722】

まず、ノード13がノード11宛のユニキャストフレームに、タググループ41のタグを付加してフレームを送信する。この際、ノード13におけるタググループ41のフレームの出力先ポートは、ツリー61のルートポートであるリンク22側のポートが指定されている。したがって、前記フレームはリンク22側に出力される。

#### 【0723】

この状態で、リンク21において障害が発生したとする。

#### 【0724】

ノード12は、リンク22よりフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、出力ポートとしてリンク21側のポート、予備出力ポートとしてリンク24側のポートを得る。そして、リンク21側に前記受信フレームを出力しようとするが、リソースモニタよりリンク21の障害検出情報を受信しているため、予備出力ポートであるリンク24側に前記受信フレームを出力する。

#### 【0725】

ノード15は、リンク24よりフレームを受信すると、タググループ41をキーとしてフォワーディングテーブルを検索し、出力ポートとしてリンク26側のポート、予備出力ポートとしてリンク24側のポートを得る。そして、リンク26側に障害が発生していないことを確認し、リンク26側に前記受信フレームを出力する。

#### 【0726】

ノード14は、リンク26よりフレームを受信すると、タググループ41をキ

ーとしてフォワーディングテーブルを検索し、出力ポートとしてリンク23側のポート、予備出力ポートとして初期値を得る。そして、リンク23側に障害が発生していないことを確認し、リンク23側に前記受信フレームを出力する。

【0727】

ノード11は、リンク23よりフレームを受信すると、自ノード宛であることを確認し、タグ削除器113にフレームを転送する。

【0728】

以上の動作により、リンク21において障害が発生した場合、あらかじめ設定されている予備出力ポートを参照し、ノード13からノード11へのユニキャストフレームを転送する場合のルートとして、リンク22、リンク23、リンク24、リンク26、リンク23を経由するルートがただちに選ばれ、ノード11へのフレーム転送が継続される。したがって高速な障害時の迂回が可能であり、ネットワークの輻輳を回避できる。

【0729】

次に、上述した第10の実施の形態の効果について説明する。

【0730】

従来、宛先への最低コスト経路が必ずしも選択されることは限らなかったが、本実施の形態では、宛先がルートノードとなるツリーを用いてユニキャストフレームを転送することにより、宛先への最低コスト経路を選択することができる。

【0731】

また従来、リンク利用率が低い一方、ルートノード付近に負荷が集中するという問題があったが、本実施の形態では、ルートノードの異なるスパニングツリーを複数系統設定することにより、リンク利用率を高め、ルートノード付近に負荷を集中させずに分散させることができる。

【0732】

さらに従来、ルートノード障害時のツリー構築に時間がかかり、その間ネットワークが停止するという問題があったが、本実施の形態では、宛先がルートノードとなるツリーを用いてフレームを転送することにより、ルートノードが宛先となるフレーム以外のフレームが、ルートノード障害の影響により長時間転送不能

となることがないため、ルートノード障害によるネットワーク停止を回避可能である。また、この結果、輻輳発生の可能性を下げることが可能である。

【0733】

さらに従来、ルートポート側リンク障害時の出力先ポートの切替に時間がかかり、その間、フレーム転送が停止するという問題があったが、本実施の形態では、あらかじめ出力リンク障害時のための予備出力リンクをフォワーディングテーブルに設定しておくことで、ルートポート側リンク、すなわち出力リンクの障害時において、高速な経路変更が可能である。この結果、輻輳発生の可能性を下げることが可能である。

【0734】

本発明のネットワークにおけるスパンニングツリー構成ノードの構成要素である各手段の機能については、それをハードウェア的に実現することは勿論として、上記した各手段の機能を実行するスパンニングツリー再構成プログラム（アプリケーションプログラム）950をコンピュータ処理装置のメモリにロードしてコンピュータ処理装置を制御することで実現することができる。このスパンニングツリー再構成プログラム950は、磁気ディスク、半導体メモリその他の記録媒体に格納され、その記録媒体からコンピュータ処理装置にロードされ、コンピュータ処理装置の動作を制御することにより、上述した各機能を実現する。

【0735】

以上好ましい実施の形態および実施例を挙げて本発明を説明したが、本発明は必ずしも上記実施の形態及び実施例に限定されるものではなく、その技術的思想の範囲内において様々に変形して実施することができる。

【0736】

【発明の効果】

以上説明したように、本発明によれば、以下のような効果が達成される。

【0737】

第1に、輻輳発生の確率を下げ、輻輳によるフレームの到着遅れや欠落が発生する頻度を減らすことができる。その理由は、構成変更前のスパンニングツリーを運用したまま、新規追加ノードを含めたスパンニングツリーを生成し、新規スパン

ングツリーの安定後に利用するスパニングツリーを切り替え、さらに、ルートノードの異なるスパニングツリーを複数系統設定するからである。

## 【0738】

第2に、ネットワークを停止させずに、スパニングツリーに属するノードの追加および削除等のスパニングツリー再構成できる。その理由は、構成変更前のスパニングツリーを運用したまま、新規追加ノードを含めたスパニングツリーを生成し、新規スパニングツリーの安定後に利用するスパニングツリーを切り替えるからである。

## 【0739】

第3に、トライフィックの負荷分散ができる。その理由は、リンクコストを、空き帯域やサーバ負荷等の動的情報によって計算するからである。

## 【0740】

第4に、経路変更にともなうスパニングツリー再構成のために、ネットワークを停止させることなく負荷分散ができる。その理由は、変更前のツリーを運用したまま、コスト変更後のツリーを生成し、新規ツリーの安定後に利用するツリーを切り替えるからである。

## 【0741】

第5に、宛先への最低コスト経路を選択することができる。その理由は、宛先がルートノードとなるツリーを用いてフレームを転送するからである。

## 【0742】

第6に、リンク利用率を高め、ルートノード付近に負荷を集中させずに分散可能である。その理由は、ルートノードの異なるスパニングツリーを複数系統設定するからである。

## 【0743】

第7に、ルートノード障害によるネットワーク停止を回避可能である。その理由は、宛先がルートノードとなるツリーを用いてフレームを転送することにより、ルートノードが宛先となるフレーム以外のフレームが、ルートノード障害の影響により長時間転送不能となることがないからである。

## 【0744】

第8に、スパンニングツリーがIEEE802.1D利用区間を通過して設定されるのを防ぎ、障害時の切替やルート変更を高速化し、輻輳発生の可能性やフレームの欠落の可能性を低下させることが可能である。その理由は、IEEE802.1D利用区間のコストを大きく設定し、スパンニングツリーがIEEE802.1D利用区間を通過して設定されるのを防ぐからである。

## 【0745】

第9に、迂回操作を高速に行い、輻輳の発生やフレームの欠落を防止できる。その理由は、IEEE802.1D利用区間の数だけツリーマネージャを作成し、相異なる1つのIEEE802.1D利用区間のコストを大きく設定したツリーをIEEE802.1D利用区間分作成し、障害等により前記区間を迂回する必要が生じた場合は、前記区間に對して大きなコストを振ったツリーを利用するよう切り替えるからである。

## 【0746】

第10に、従来、スパンニングツリープロトコルで利用されているHELLOフレームの送信間隔が長かったため、高速な障害検出ができなかつたが、短い間隔で障害検出用フレームを送受信する障害検出器を追加することにより、HELLOフレームよりも高速な障害検出が可能となる。また、この結果、輻輳発生およびフレーム欠落の可能性を下げることが可能となる。

## 【0747】

第11に、従来、ブロードキャスト時に、宛先への最低コスト経路が必ずしも選択されるとは限らなかつたが、本発明では、送信元ノードがルートノードとなるツリーを用いてブロードキャストフレームを転送することにより、全ノードへの最低コスト経路を選択してブロードキャストフレームを転送することができる。

## 【0748】

第12に、送信元ノードがルートノードとなるツリーを用いてブロードキャストフレームを転送することにより、ルートノードが送信元ノードとなるフレーム以外のブロードキャストフレームが、ルートノード障害の影響により長時間転送不能となることがないため、ルートノード障害によるネットワーク停止を回避可

能である。また、この結果、輻輳発生の可能性を下げることが可能である。

【0749】

第13に、従来、宛先への最低コスト経路が必ずしも選択されるとは限らなかったが、宛先がルートノードとなるツリーを用いてユニキャストフレームを転送することにより、宛先への最低コスト経路を選択することができる。

【0750】

第14に、従来、ルートポート側リンク障害時の出力先ポートの切替に時間がかかり、その間、フレーム転送が停止するという問題があったが、本発明では、あらかじめ出力リンク障害時のための予備出力リンクをフォワーディングテーブルに設定しておくことで、ルートポート側リンク、すなわち出力リンクの障害時ににおいて、高速な経路変更が可能である。この結果、輻輳発生の可能性を下げることが可能である。

【図面の簡単な説明】

【図1】 従来のVLANタグ付きイーサネット(R)フレームの構成例を示す図である。

【図2】 本発明の拡張タグ付きイーサネット(R)フレームの構成例を示す図である。

【図3】 本発明の拡張タグ付きイーサネット(R)フレームの他の構成例を示す図である。

【図4】 本発明の拡張タグ格納領域の構成例を示す図である。

【図5】 本発明におけるConfiguration BPDUフレームのフレーム構成を示すフォーマット図である。

【図6】 本発明におけるTopology Change Notification BPDUフレームのフレーム構成を示すフォーマット図である。

【図7】 本発明の第1の実施の形態の構成を示すブロック図である。

【図8】 本発明の第1の実施の形態における、ノード11の構成を示すブロック図である。

【図9】 本発明の第1の実施の形態における、フォワーディングテーブル114の構成例を示す表である。

【図10】 本発明の第1の実施の形態における、ツリーマネージャ115  
1の構成を示すブロック図である。

【図11】 本発明の第1の実施の形態における、ツリーセレクタ116の  
構成を示すブロック図である。

【図12】 本発明の第1の実施の形態における、メインコントローラ11  
64の動作を示す流れ図である。

【図13】 本発明の第1の実施の形態における、タグテーブル117の構  
成例を示す表である。

【図14】 本発明の第1の実施の形態における、ノード700が追加され  
る前

のスパニングツリー51の構成を示すブロック図である。

【図15】 本発明の第1の実施の形態における、ノード700が追加され  
た後

のスパニングツリー52の構成を示すブロック図である。

【図16】 本発明の第1の実施の形態における、制御フレームのやりとり  
を示すシーケンス図である。

【図17】 本発明の第2の実施の形態における、ツリーセレクタ116の  
構成

を示すブロック図である。

【図18】 本発明の第2の実施の形態における、メインコントローラ11  
64

の動作を示す流れ図である。

【図19】 本発明の第3の実施の形態における、メインコントローラ11  
64の動作を示す流れ図である。

【図20】 本発明の第4の実施の形態における、ノード11の構成を示す  
ブロック図である。

【図21】 本発明の第4の実施の形態における、フォワーディングテーブ  
ル114の構成例を示す表である。

【図22】 本発明の第4の実施の形態における、タグテーブル117の構

成例を示す表である。

【図23】 本発明の第4の実施の形態における、ツリー61の構成を示す  
ブロック図である。

【図24】 本発明の第4の実施の形態における、ツリー62の構成を示す  
ブロック図である。

【図25】 本発明の第4の実施の形態における、ツリー63の構成を示す  
ブロック図である。

【図26】 本発明の第4の実施の形態における、ツリー64の構成を示す  
ブロック図である。

【図27】 本発明の第4の実施の形態における、ツリー65の構成を示す  
ブロック図である。

【図28】 本発明の第4の実施の形態における、ツリー66の構成を示す  
ブロック図である。

【図29】 本発明の第4の実施の形態の構成を示すブロック図である。

【図30】 本発明の第4の実施の形態における、ツリー74の構成を示す  
ブロック図である。

【図31】 本発明の第5の実施の形態における、ツリーマネージャ115  
1の構成を示すブロック図である。

【図32】 本発明の第5の実施の形態における、ツリー71の構成を示す  
ブロック図である。

【図33】 本発明の第5の実施の形態における、ツリー72の構成を示す  
ブロック図である。

【図34】 本発明の第5の実施の形態における、ツリー73の構成を示す  
ブロック図である。

【図35】 本発明の第6の実施の形態における、ノード11の構成を示す  
ブロック図である。

【図36】 本発明の第6の実施の形態における、ツリー67の構成を示す  
ブロック図である。

【図37】 本発明の第6の実施の形態における、ツリー68の構成を示す

ブロック図である。

【図38】 本発明の第6の実施の形態における、ツリー69の構成を示す  
ブロック図である。

【図39】 本発明の第6の実施の形態における、ツリー70の構成を示す  
ブロック図である。

【図40】 本発明の第7の実施の形態における、ノード11の構成を示す  
ブロック図である。

【図41】 本発明の第8の実施の形態における、ノード11の構成を示す  
ブロック図である。

【図42】 本発明の第8の実施の形態における、フォワーディングテーブ  
ル114γの構成例を示す表である。

【図43】 本発明の第8の実施の形態における、ツリーマネージャ115  
1γの構成を示すブロック図である。

【図44】 本発明の第8の実施の形態における、ツリー61の設定状況例  
を示す表である。

【図45】 本発明の第9の実施の形態における、ノード11の構成を示す  
ブロック図である。

【図46】 本発明の第9の実施の形態における、フォワーディングテーブ  
ル114βの構成例を示す表である。

【図47】 本発明の第9の実施の形態における、ツリーマネージャ115  
1βの構成を示すブロック図である。

【図48】 本発明の第9の実施の形態における、ツリー61の設定状況例  
を示す表である。

【図49】 本発明の第10の実施の形態における、ノード11の構成を示  
すブロック図である。

【図50】 本発明の第10の実施の形態における、フォワーディングテー  
ブル114αの構成例を示す表である。

【図51】 本発明の第10の実施の形態における、ツリーマネージャ11  
51αの構成を示すブロック図である。

【図52】 本発明の第10の実施の形態における、ツリー61の設定状況例を示す表である。

【図53】 本発明における、拡張フレームの他の構成例を示す図である。

【図54】 図23に示すスパニングツリーの構成における各ノードのポート状態を説明する図である。

【符号の説明】

1 1 ~ 1 8 ノード

2 1 ~ 3 0 双方向リンク

3 1 現用系B P D U (タググループ4 1)

3 2 予備系B P D U (タググループ4 2)

3 3 現用系B P D U (タググループ4 2)

3 4 予備系B P D U (タググループ4 1)

3 5 利用タググループ変更通知 (G V R P等)

4 1 ~ 5 0 タググループ

5 1 ~ 5 2 スパニングツリー

6 1 ~ 7 3 スパニングツリー]

8 1 ~ 8 8 双方向リンク

9 1 ~ 9 7 クライアント

1 1 1 フレーム転送器

1 1 1  $\alpha$  フレーム転送器

1 1 1  $\beta$  フレーム転送器

1 1 1  $\gamma$  フレーム転送器

1 1 2 タグ挿入器

1 1 3 タグ削除器

1 1 4 フォワーディングテーブル

1 1 4  $\alpha$  フォワーディングテーブル

1 1 4  $\beta$  フォワーディングテーブル

1 1 4  $\gamma$  フォワーディングテーブル

1 1 6 ツリーセレクタ

- 1 1 7 タグテーブル
- 1 1 8 設定インターフェース
- 1 1 9 リソースモニタ
- 1 1 4 3 予備出力ポート
- 1 1 4 4 ブロードキャスト出力ポート
- 1 1 5 0 分別器
- 1 1 5 1 ツリーマネージャ
- 1 1 5 1  $\alpha$  ツリーマネージャ
- 1 1 5 1  $\beta$  ツリーマネージャ
- 1 1 5 1  $\gamma$  ツリーマネージャ
- 1 1 5 2 ツリーマネージャ
- 1 1 5 3 ツリーマネージャ
- 1 1 6 1 タグ削除器
- 1 1 6 2 GVRP送信機
- 1 1 6 3 タグ挿入器
- 1 1 6 4 メインコントローラ
- 1 1 6 5 安定タイマ
- 1 1 6 6 到着間隔タイマ
- 1 1 6 7 コスト参照タイマ
- 1 1 6 8 関数演算器
- 1 1 6 9 平滑化回路
- 1 1 5 1 1 タグ削除器
- 1 1 5 1 2 BPDU送信機
- 1 1 5 1 3 タグ挿入器
- 1 1 5 1 4 ツリーコントローラ
- 1 1 5 1 4  $\alpha$  ツリーコントローラ
- 1 1 5 1 4  $\beta$  ツリーコントローラ
- 1 1 5 1 4  $\gamma$  ツリーコントローラ
- 1 1 5 1 5 ツリーテーブル

11516 コスト操作器

11641～11649 状態

1164A 状態

1164B 状態

120 障害検出器

2800 TPID

2801 TC1

2802 Priority (プライオリティ)

2803 CFI

2804 VLAN ID

3508 ブロードキャストタグ (拡張タグ)

5001 拡張タグ識別領域

5002 拡張タグ情報領域

5003 Priority、タグタイプフィールド

5004 拡張タグ情報フィールド

6101 ノード

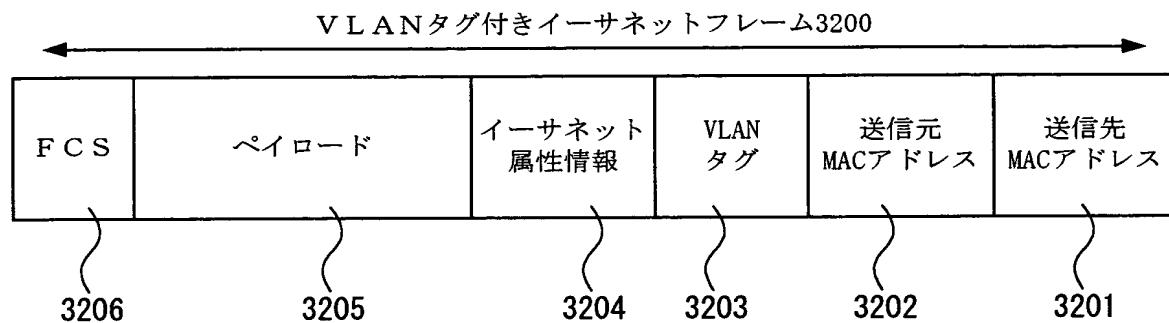
6102 ルートポート

6103 オルタネートポート

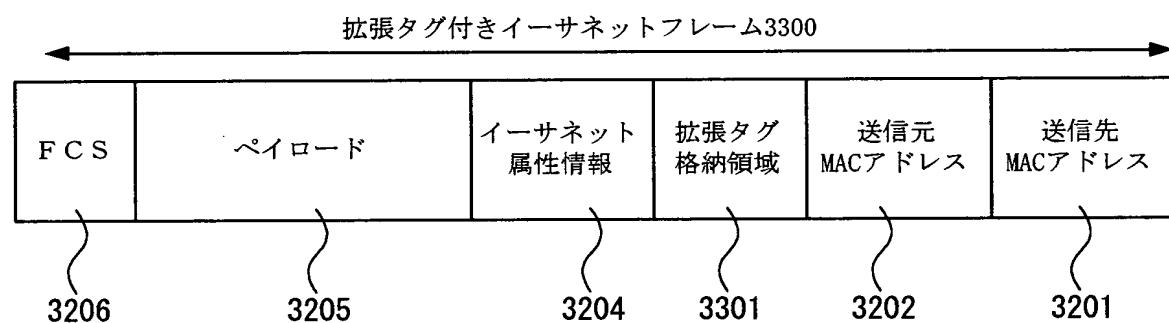
6104 ディジグネイテッドポート

【書類名】 図面

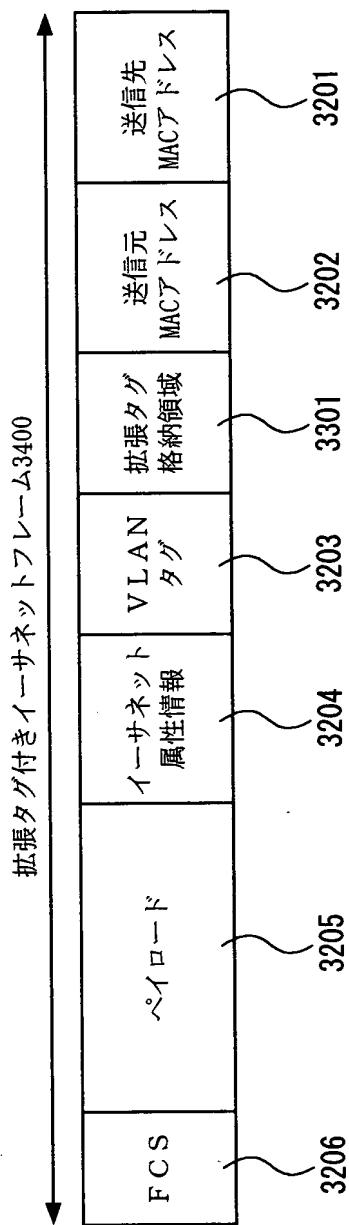
【図1】



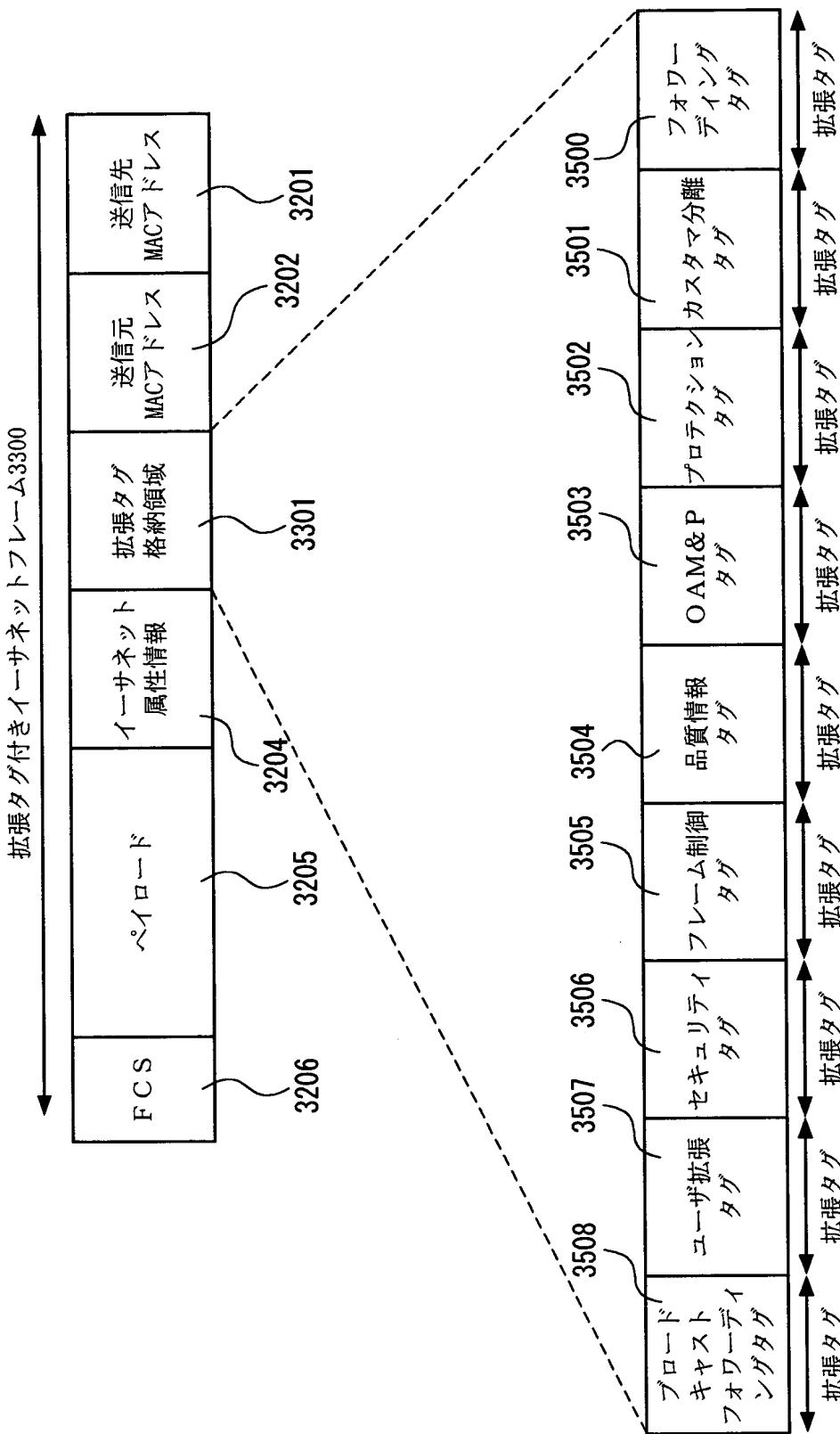
【図2】



【図3】



〔図4〕

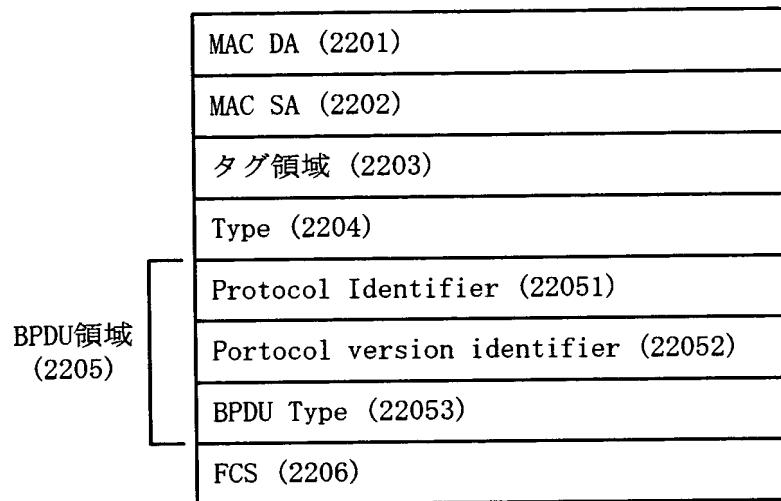


【図5】

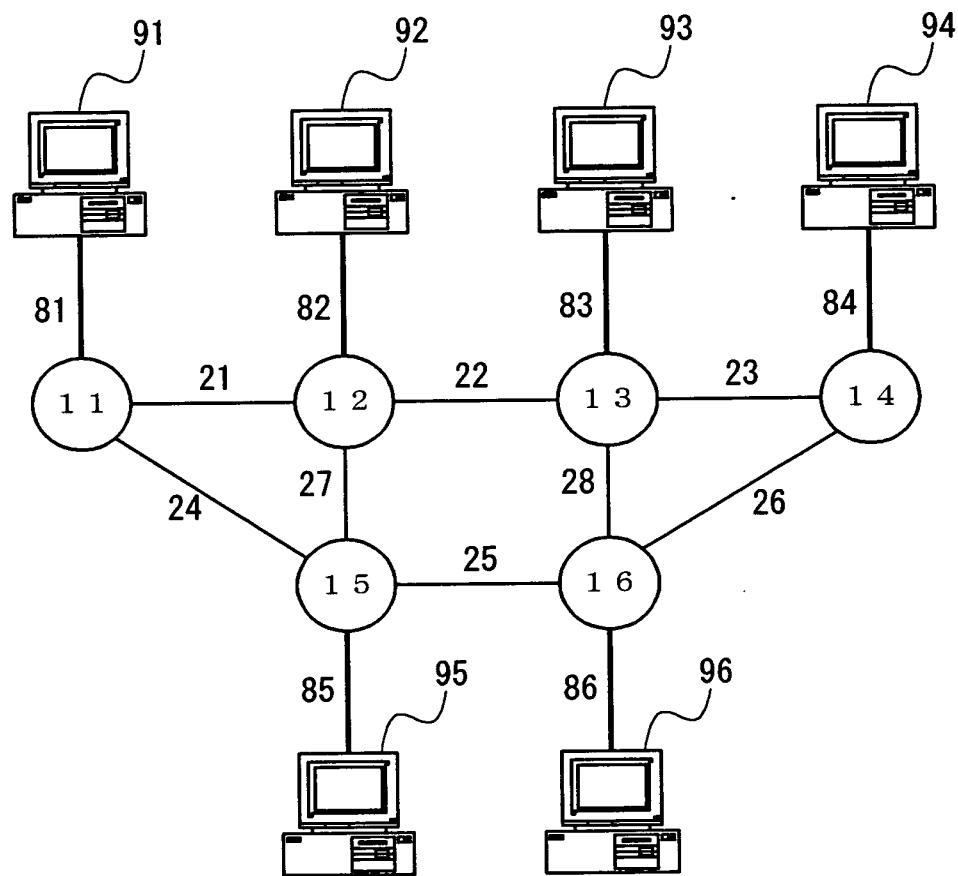
MAC DA (2201)
MAC SA (2202)
タグ領域 (2203)
Type (2204)
Protocol Identifier (22051)
Portocol Version identifier (22052)
BPDU Type (22053)
Flags (22054)
Root Identifier (22055)
Root Path Cost (22056)
Bridge Identifier (22057)
Port Identifier (22058)
Message Age (22059)
MAX Age (2205A)
Hello Time (2205B)
Forward Deley (2205C)
FCS (2206)

BPDU領域  
(2205)

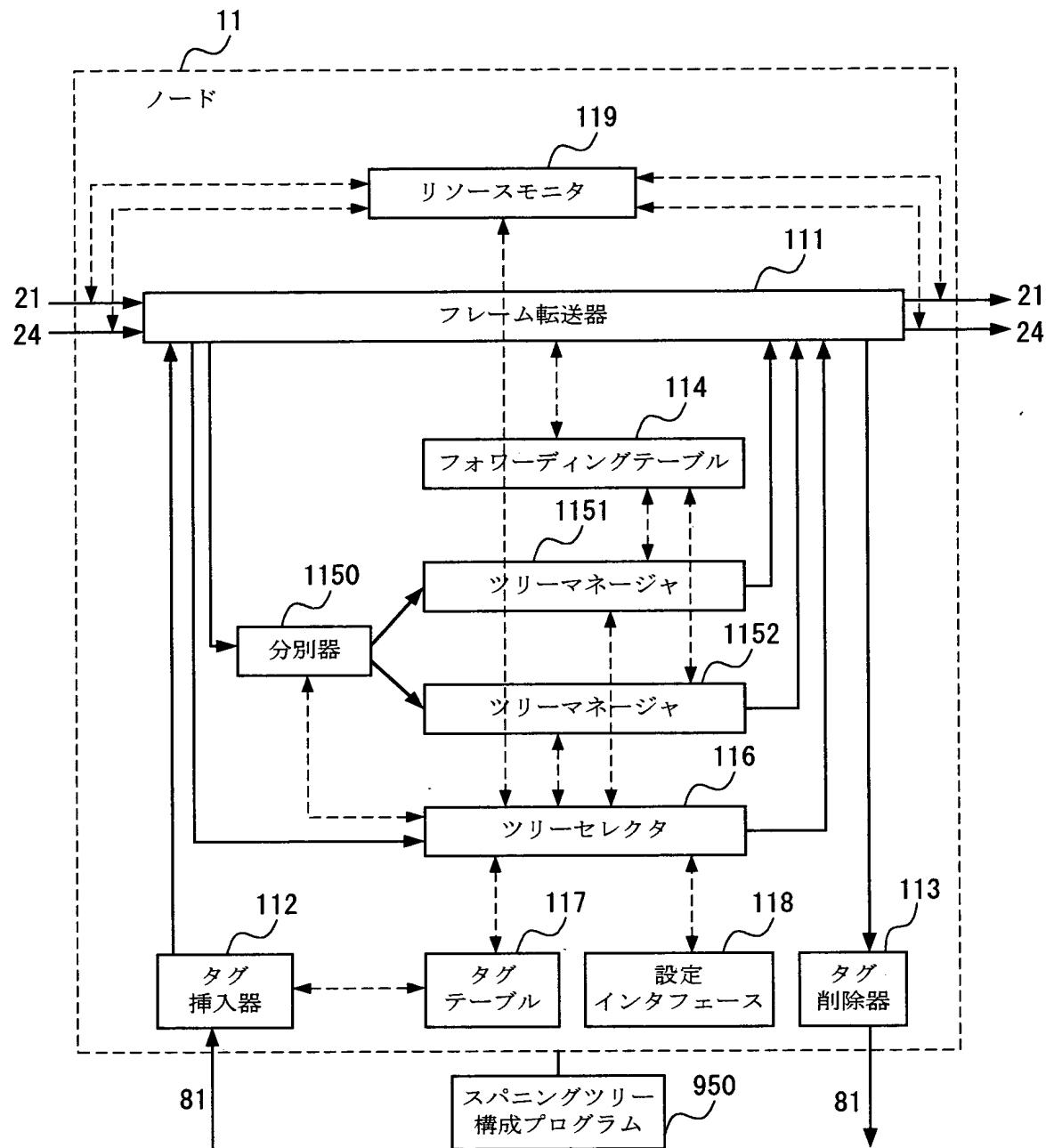
【図6】



【図7】



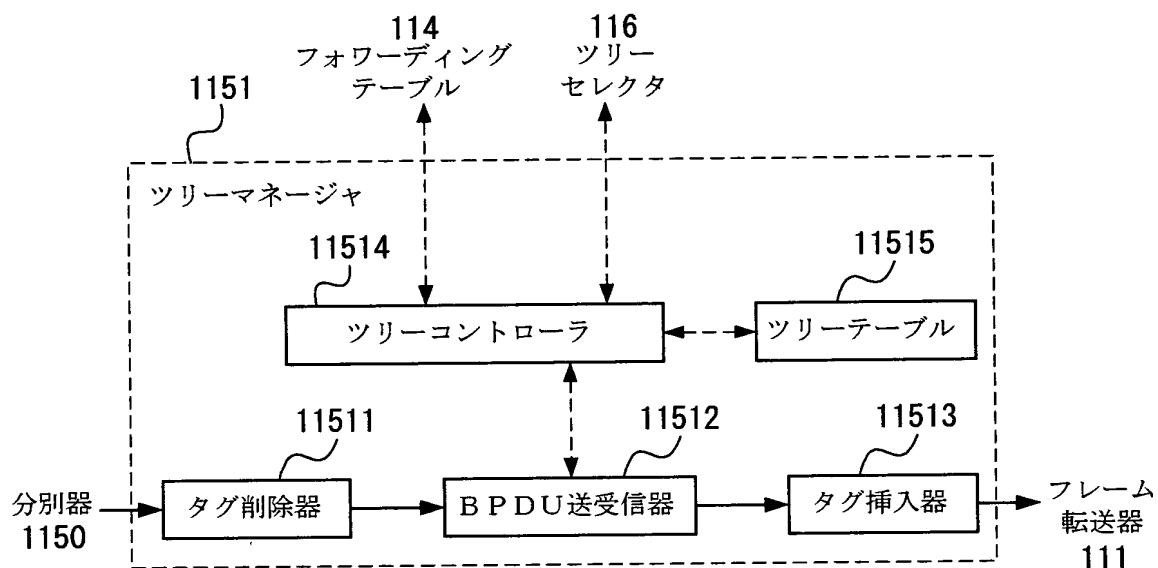
【図 8】



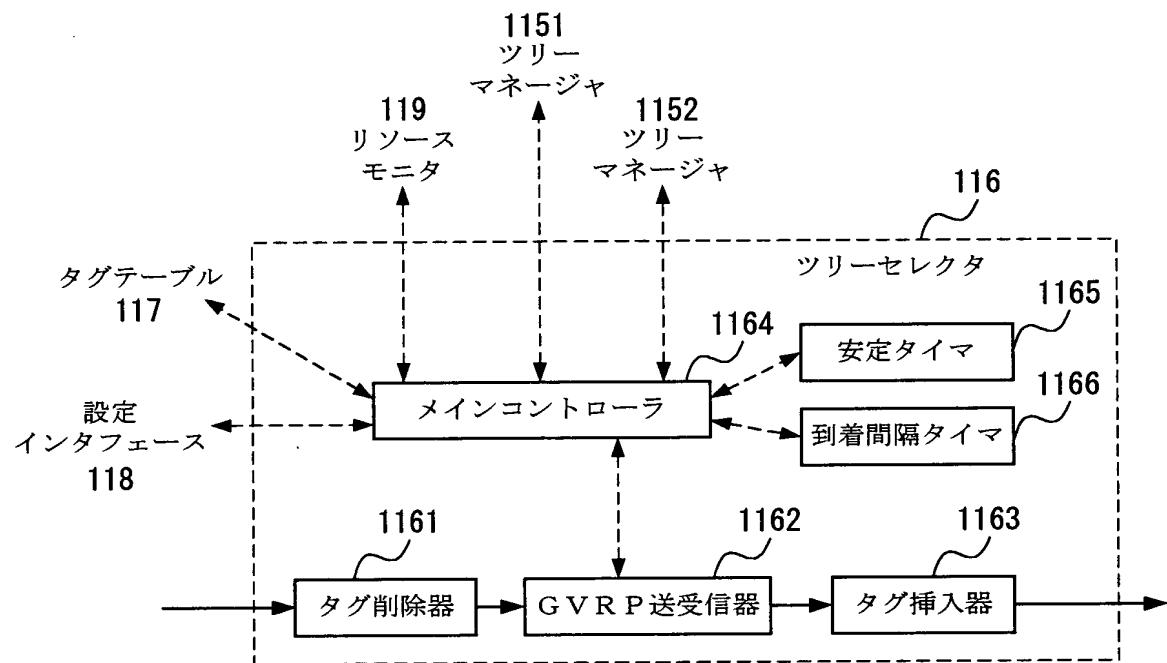
【図9】

114 フォワーディングテーブル	
タグ 1 1 4 1	出力ポート 1 1 4 2
現用系タグ	Port 2 (例)
予備系タグ	Port 1 (例)

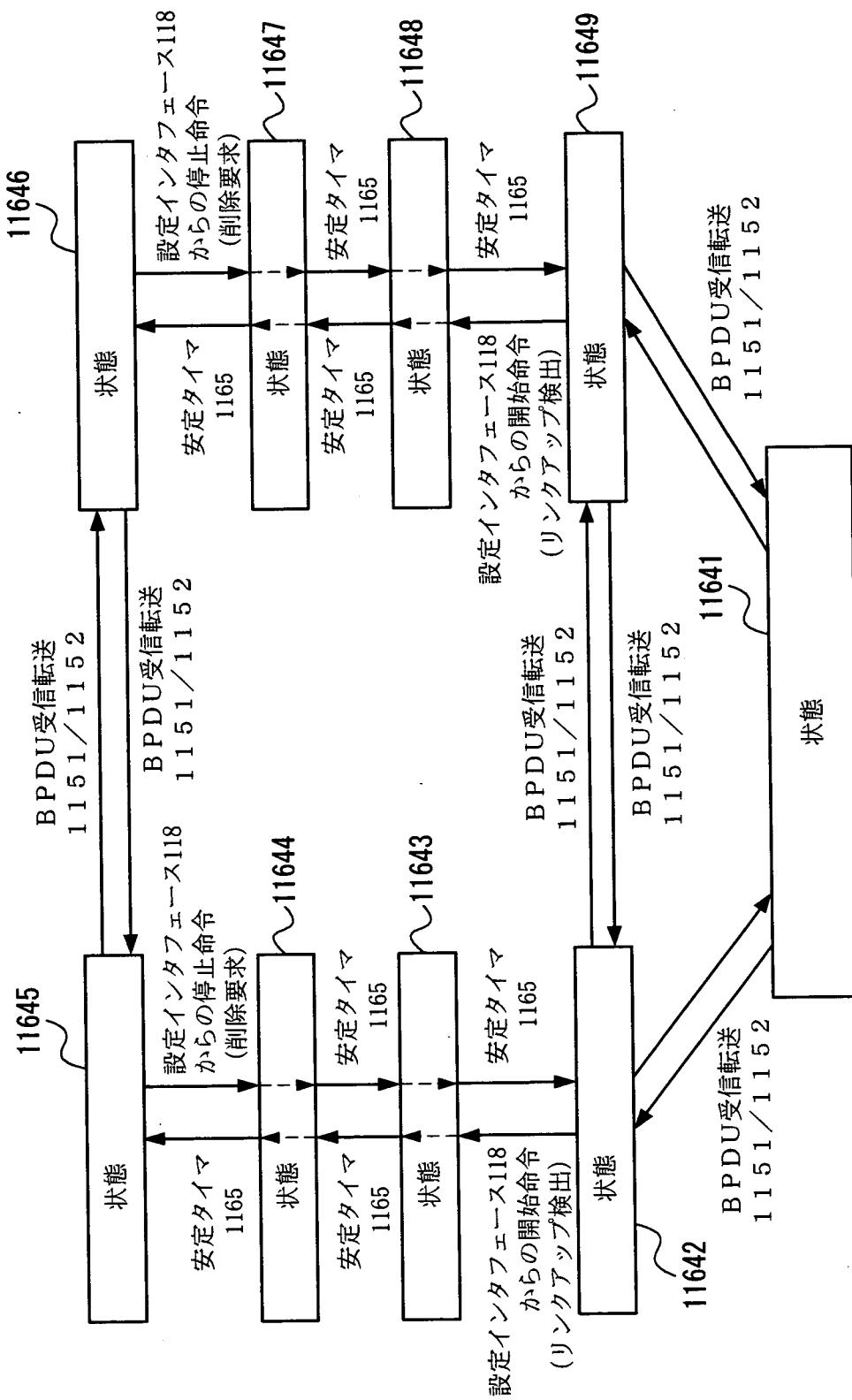
【図10】



【図11】



【図12】

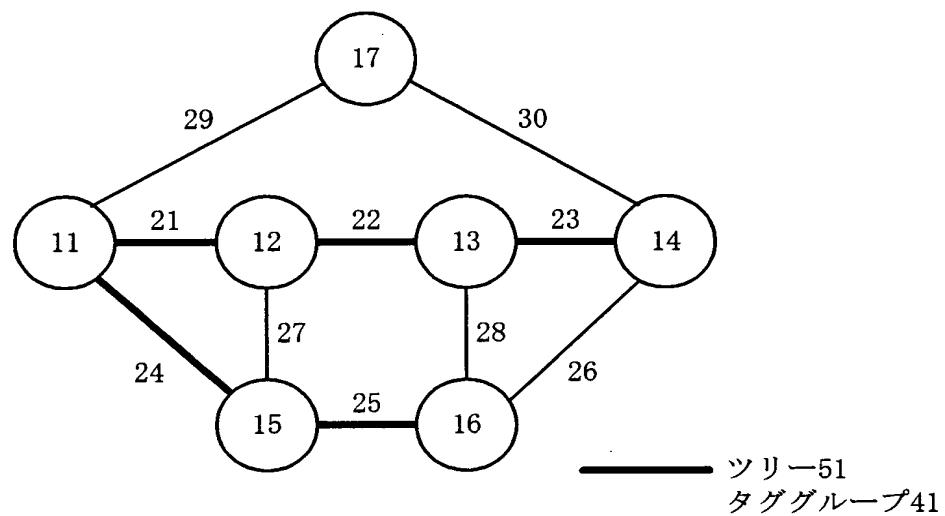


【図13】

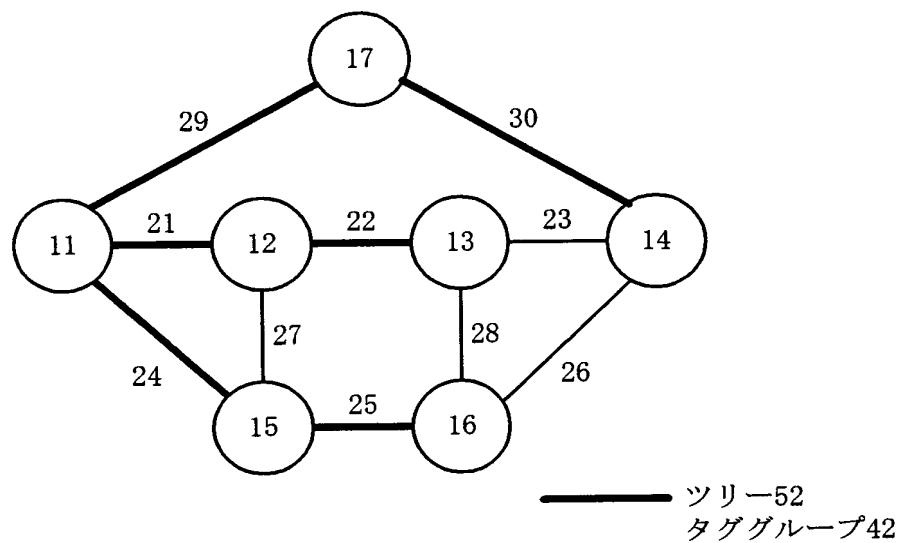
117 タグテーブル

宛先MACアドレス1171	挿入タグ1172
ネットワーク全体の全てのノードを表す値	現在、現用系となっているタグを表すタググループのID

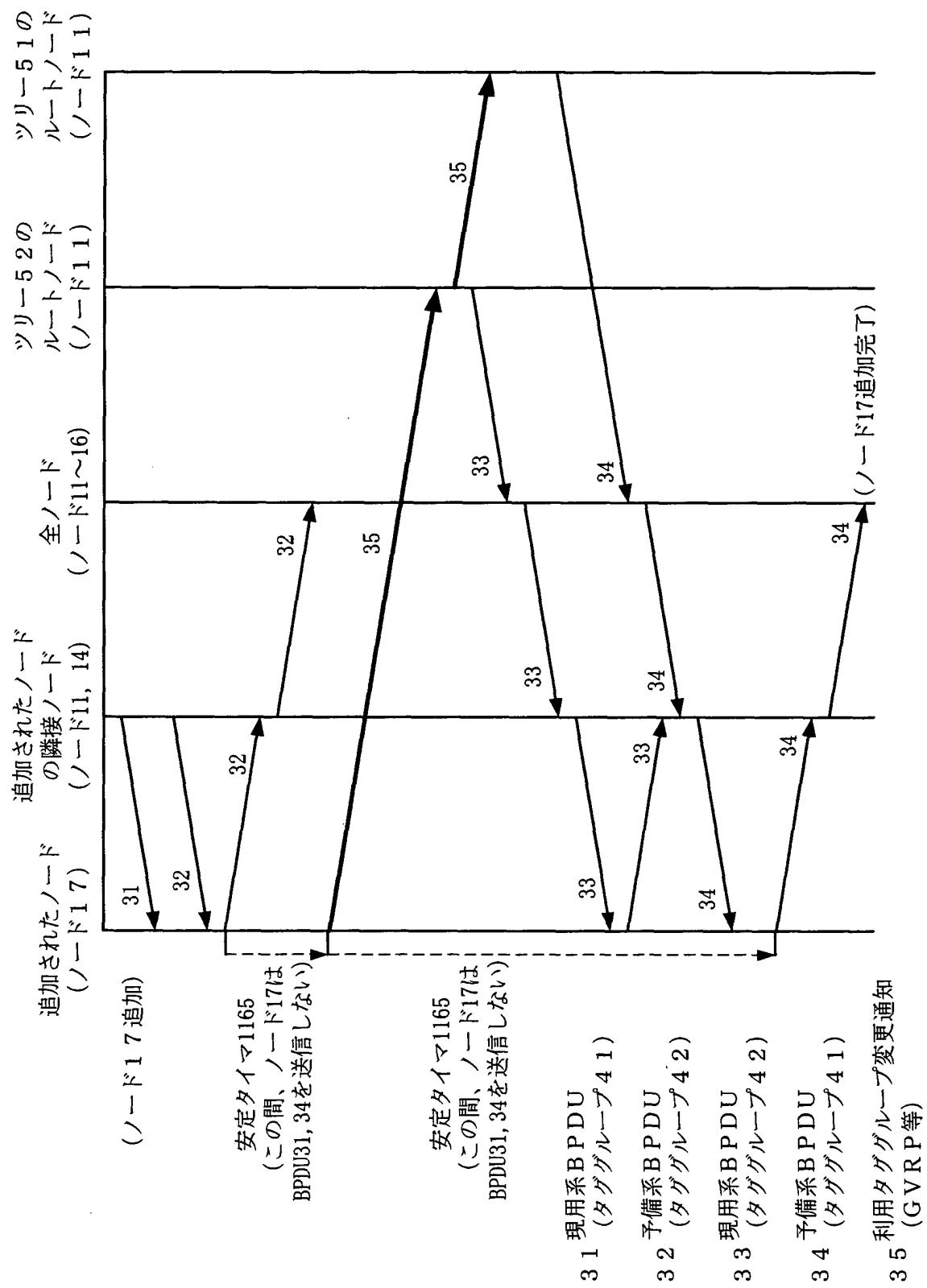
【図14】



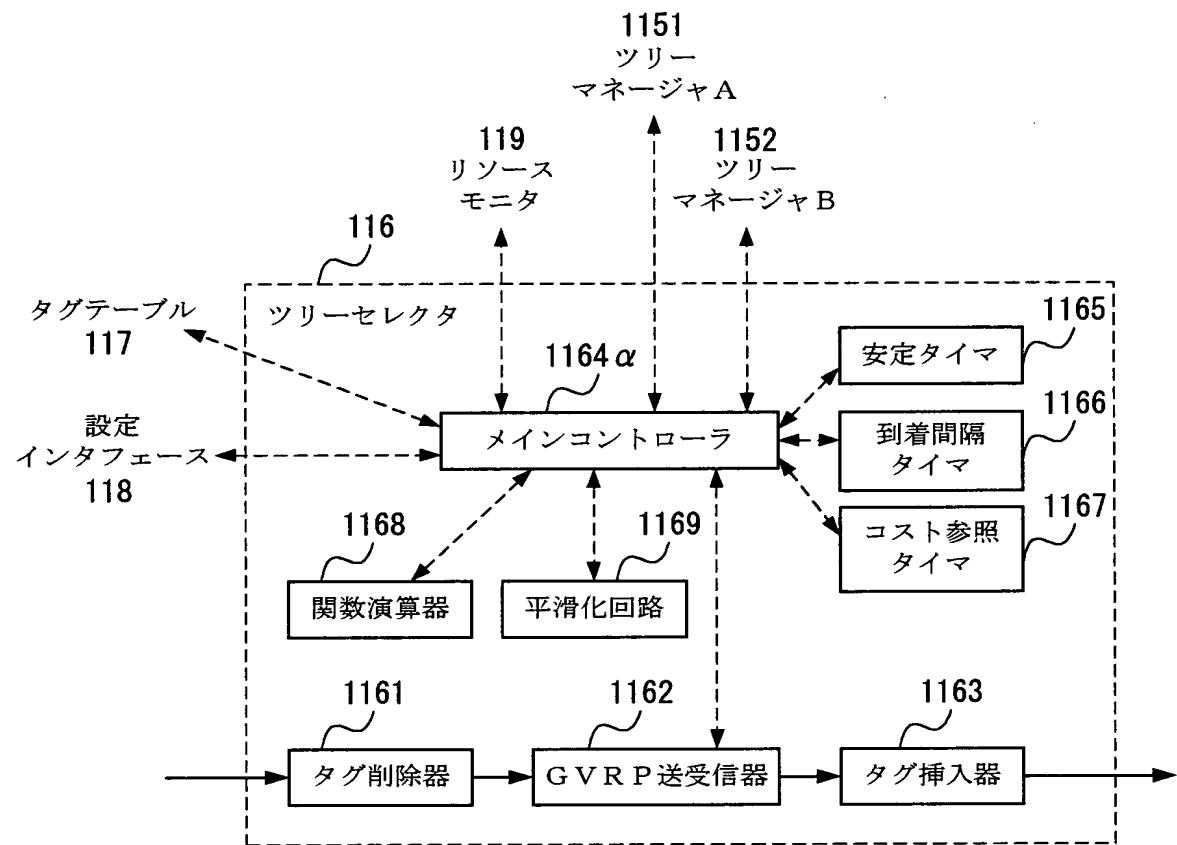
【図15】



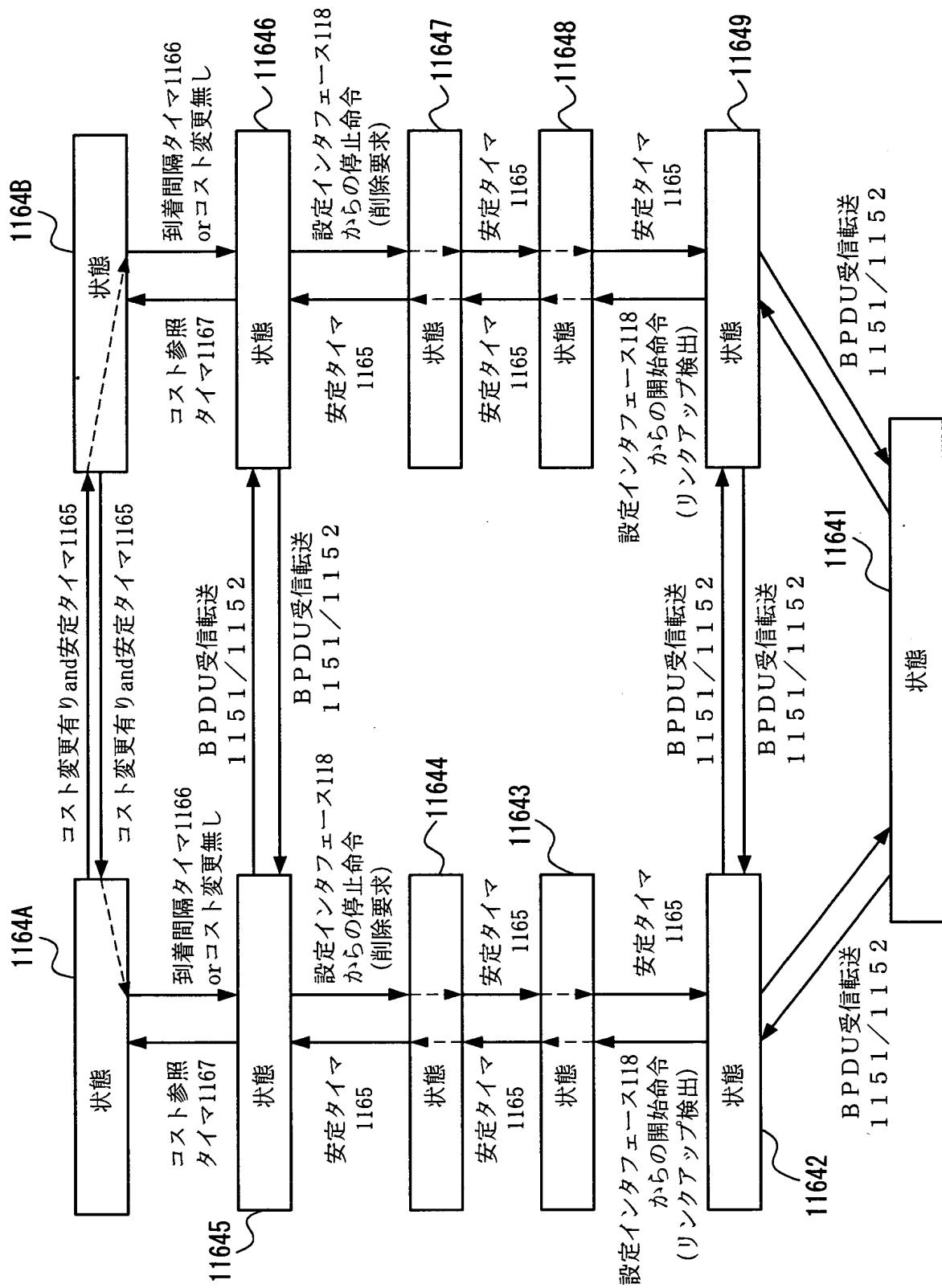
### 【図16】



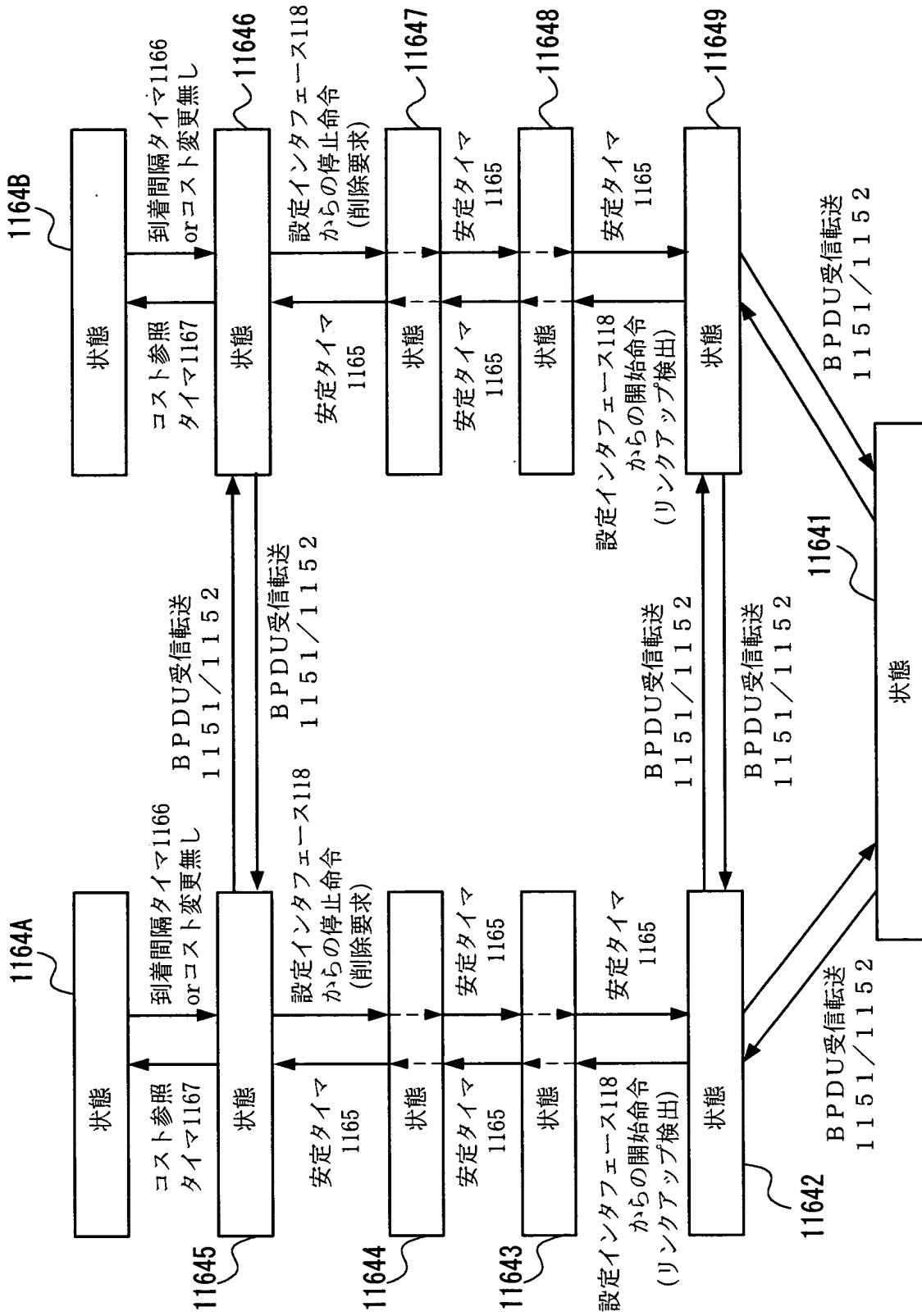
【図17】



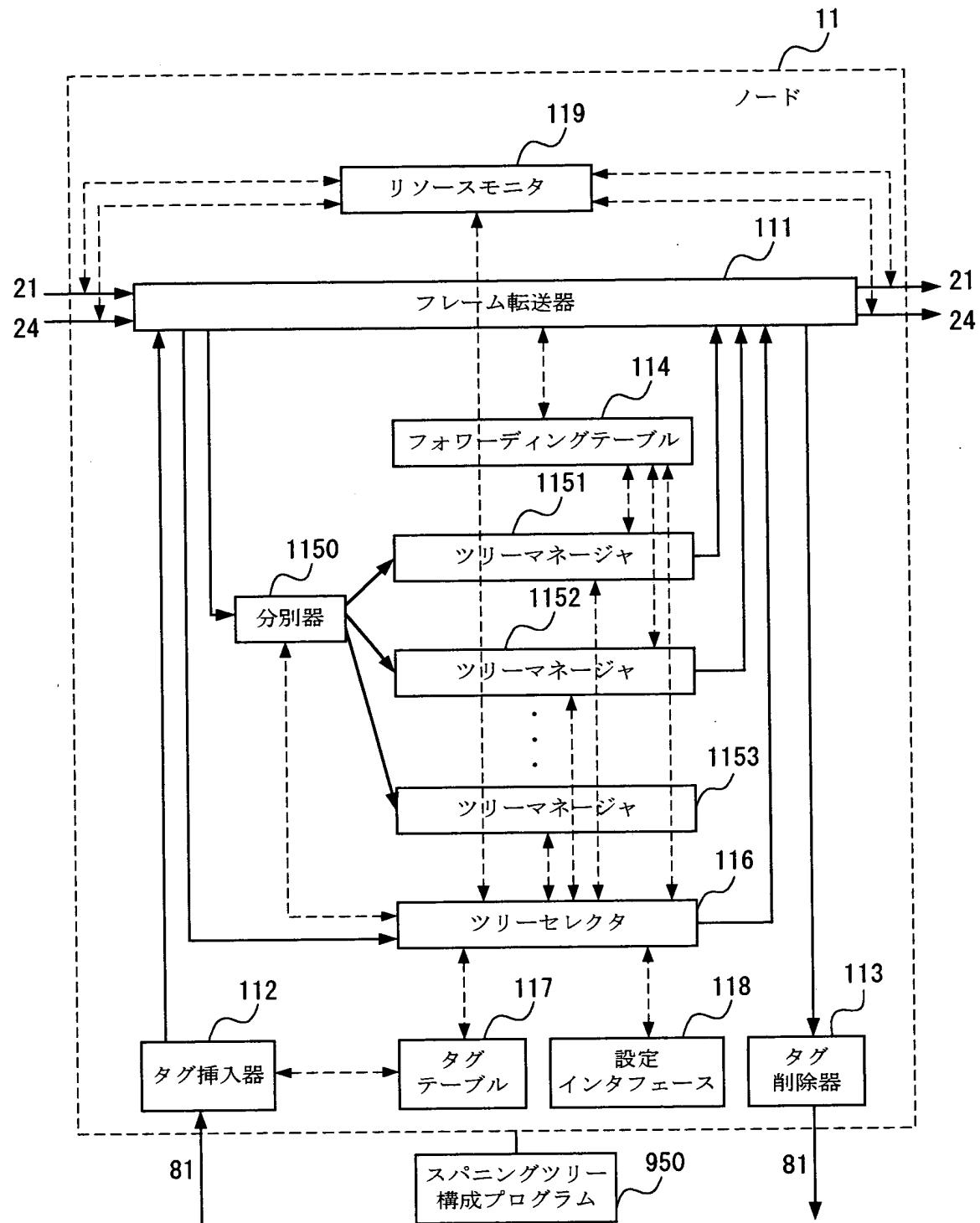
【図18】



【図19】



【図20】



【図21】

114 フォワーディングテーブル

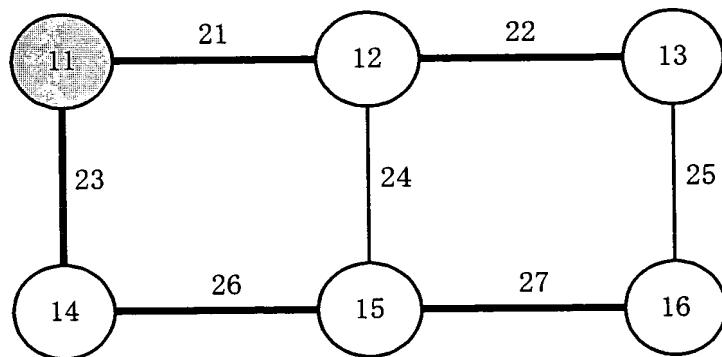
タグ 1141	出力ポート 1142
41	Port 1
42	Port 2
43	Port 2
44	Port 1

【図22】

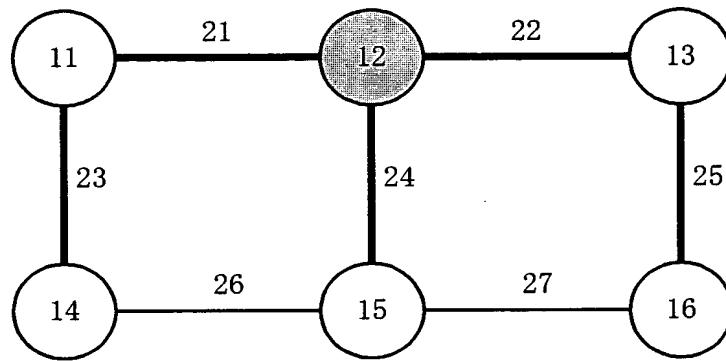
117 タグテーブル

宛先MACアドレス 1171	挿入タグ 1172
91A	41
91C	41
92B	42
93X	43

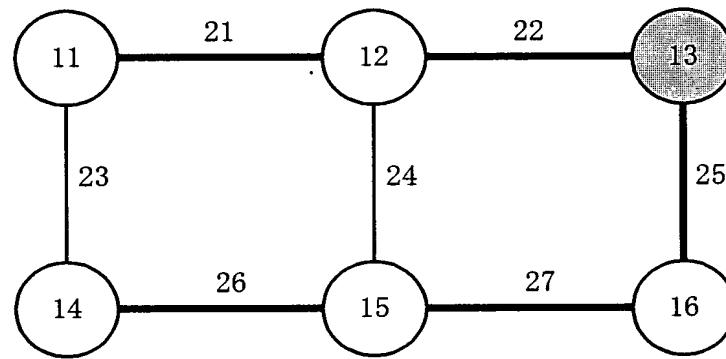
【図23】



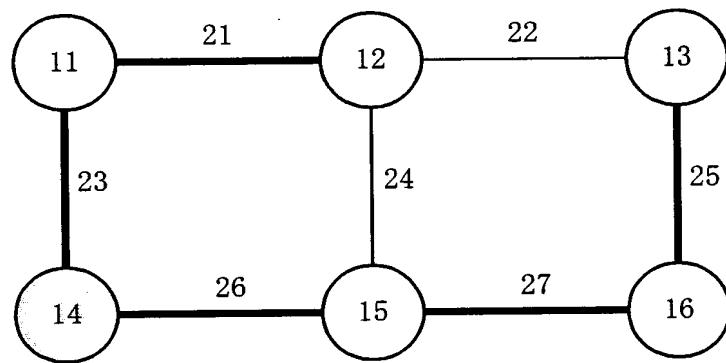
【図24】



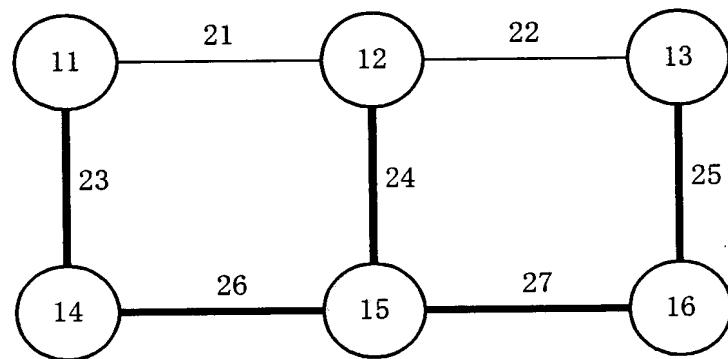
【図25】



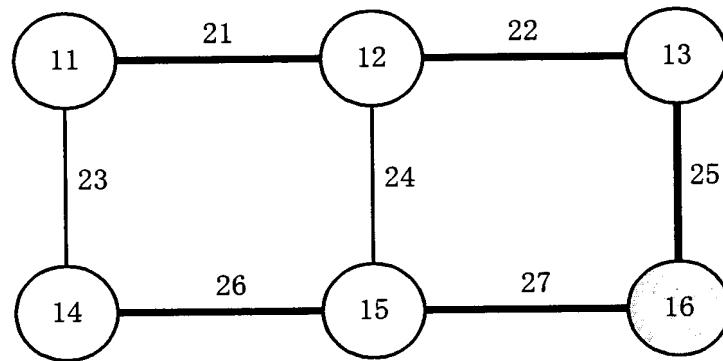
【図26】



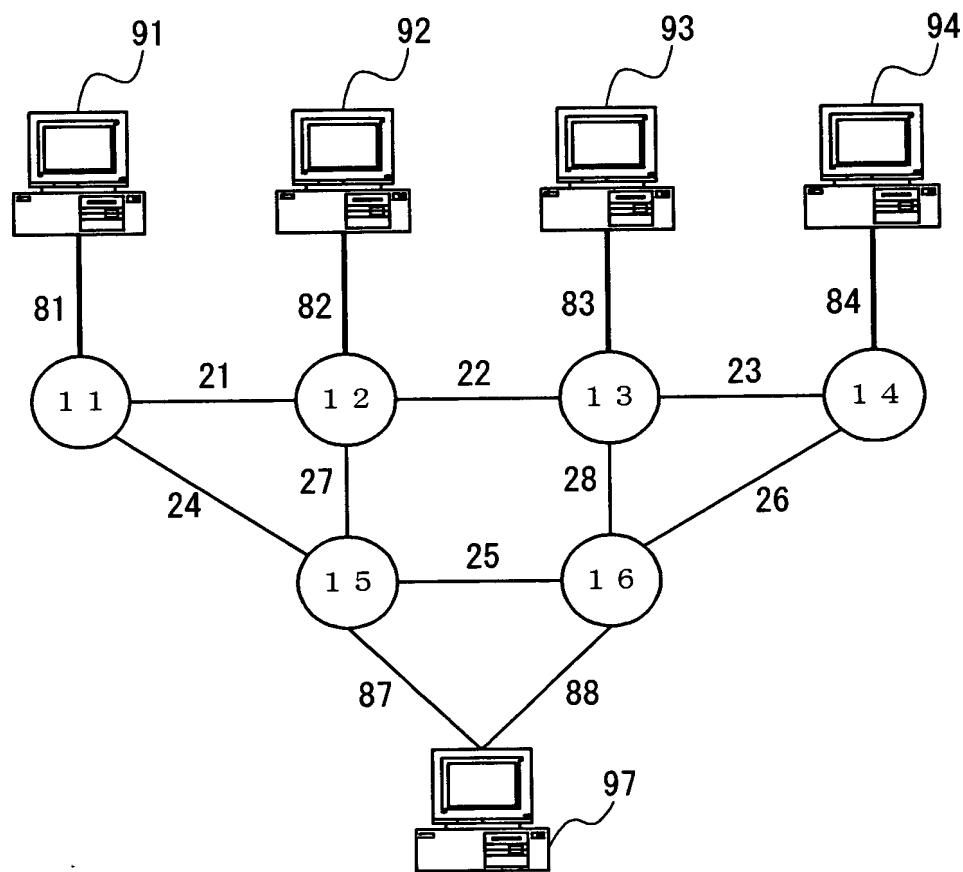
【図27】



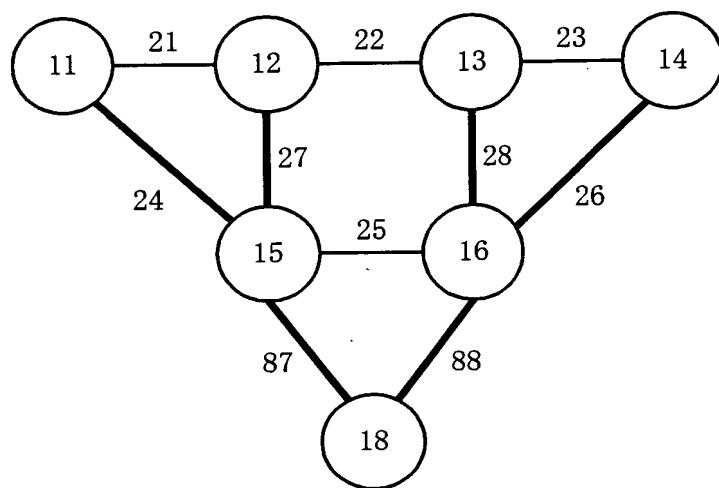
【図28】



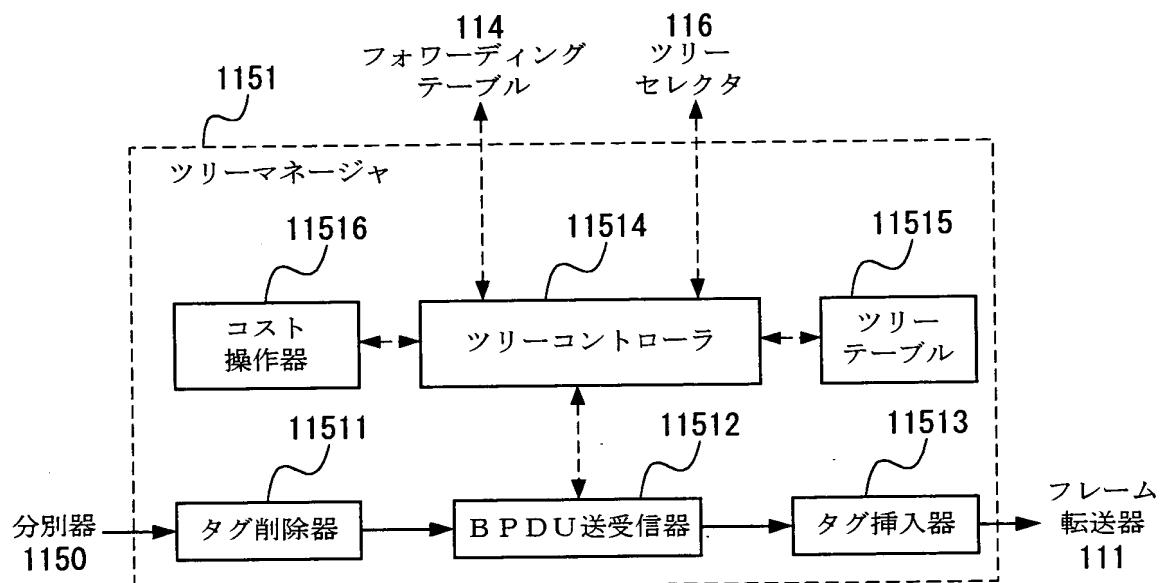
【図29】



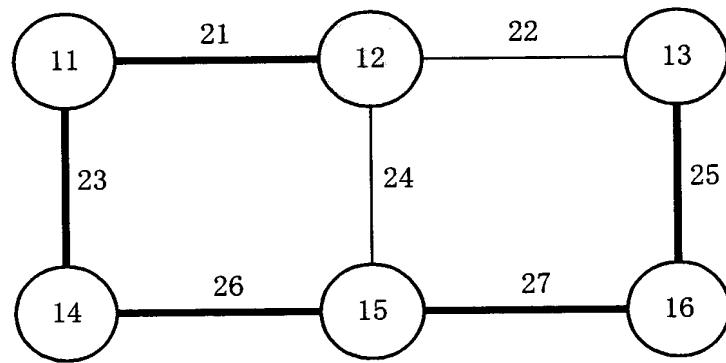
【図30】



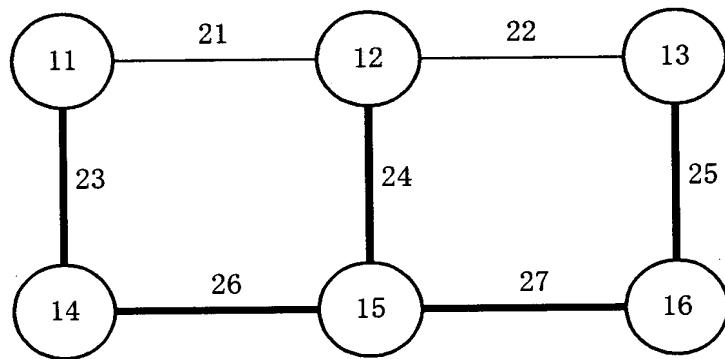
【図31】



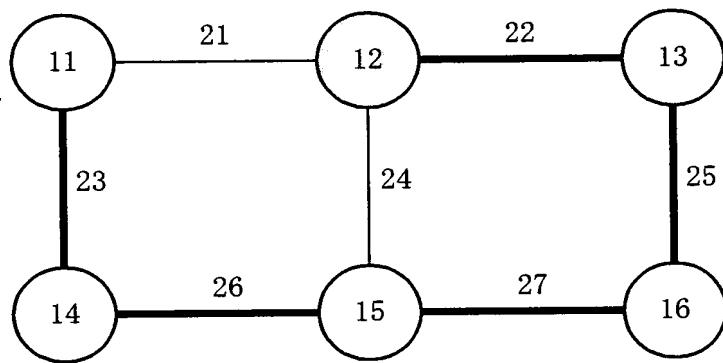
【図32】



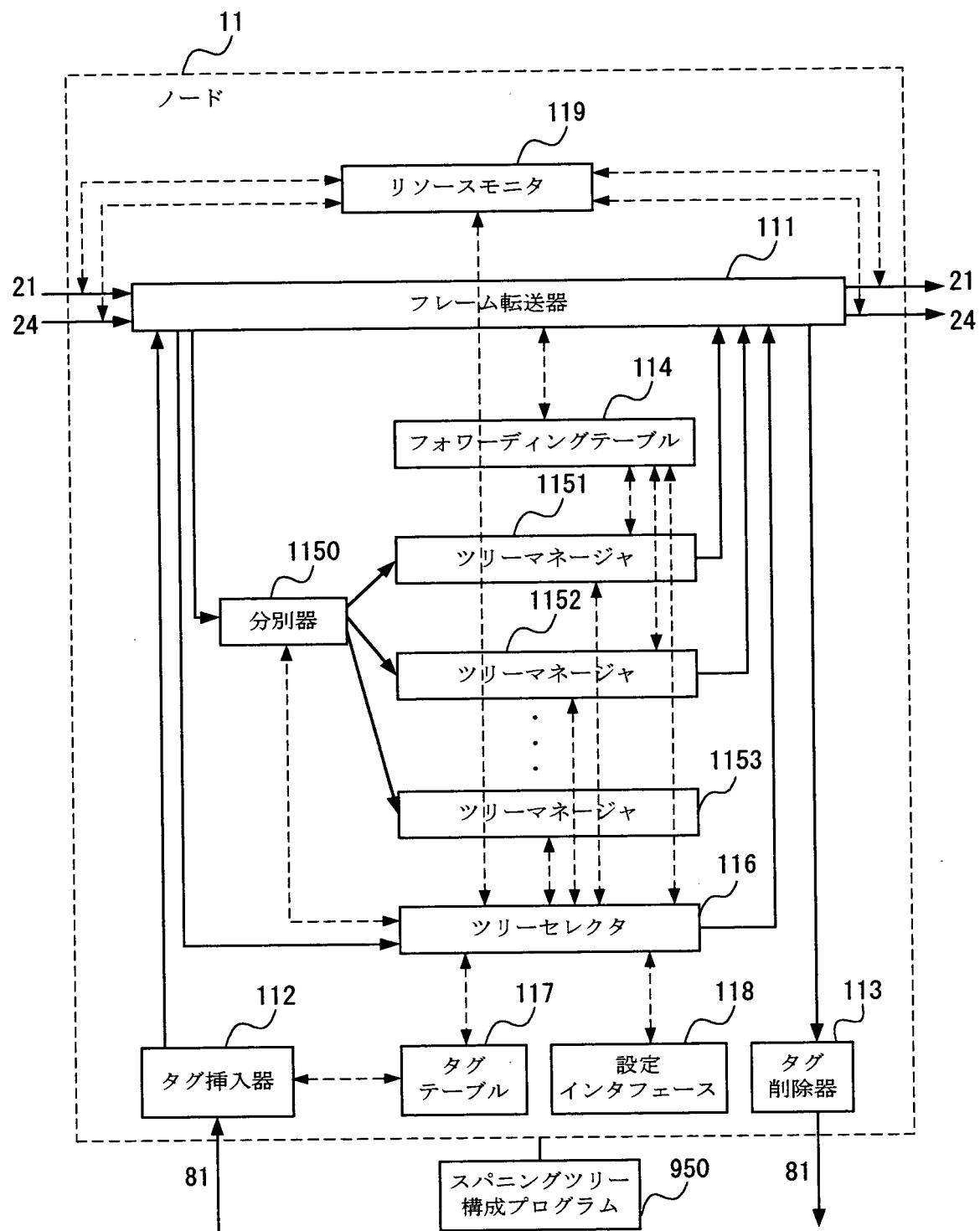
【図33】



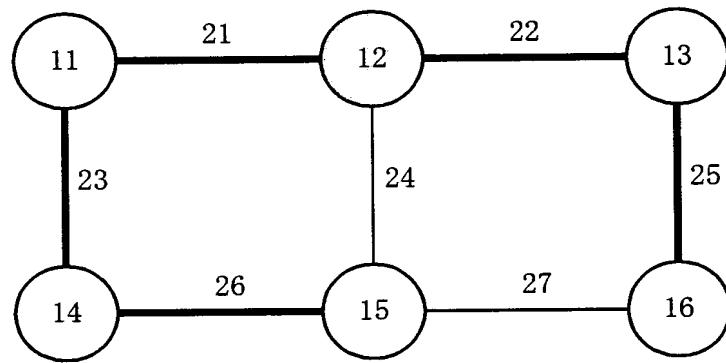
【図34】



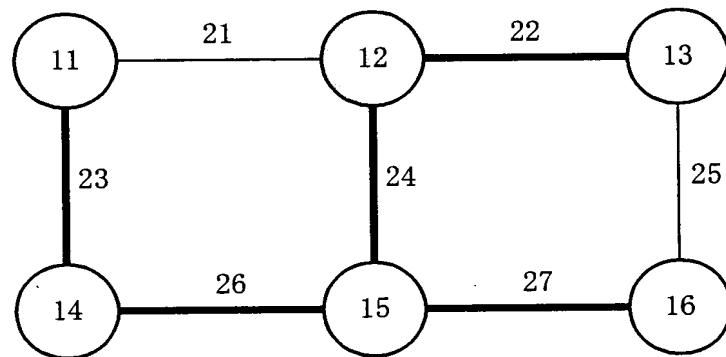
【図35】



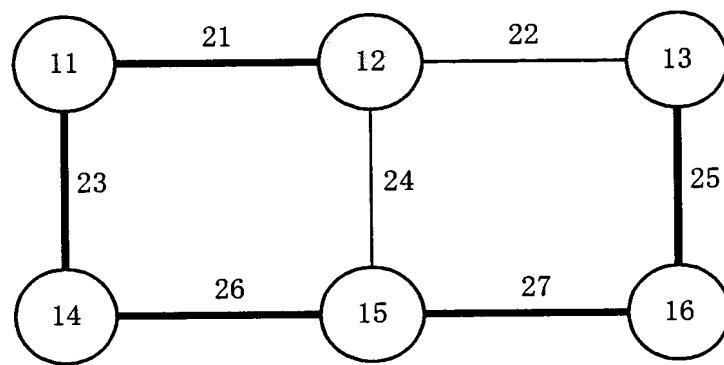
【図36】



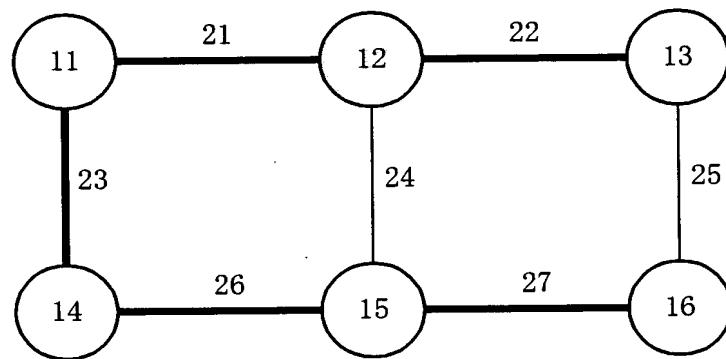
【図37】



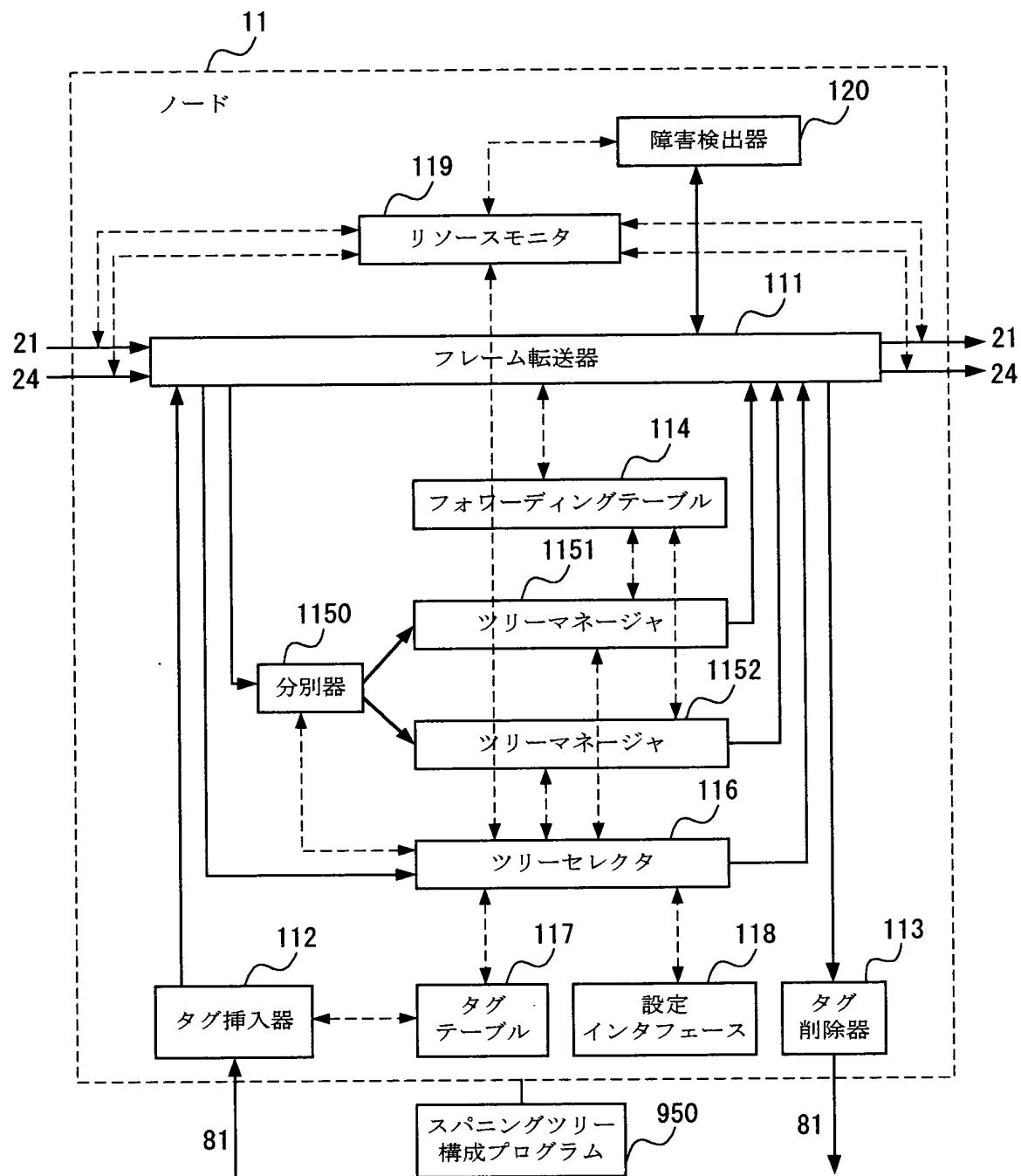
【図38】



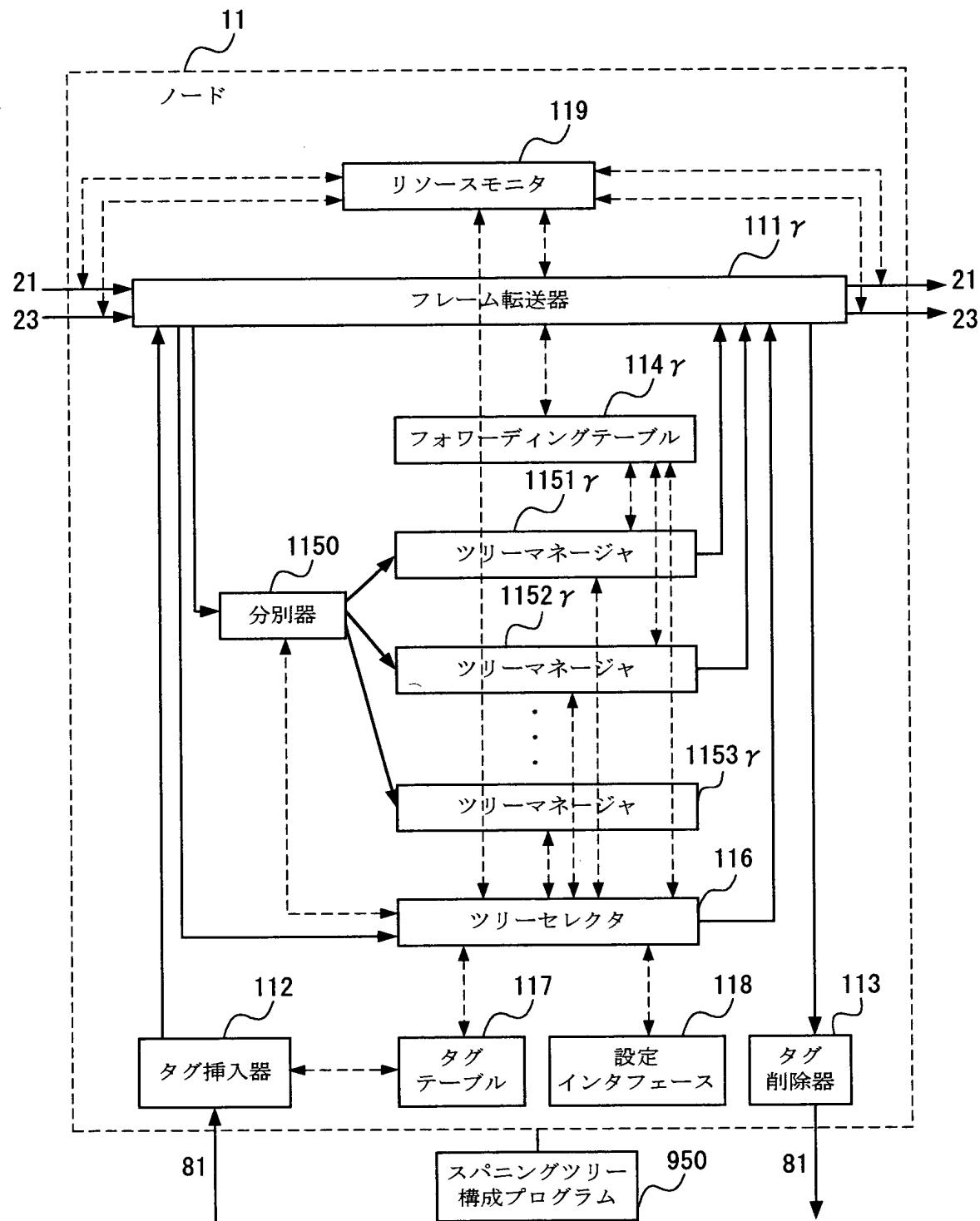
【図39】



【図40】



【図4-1】

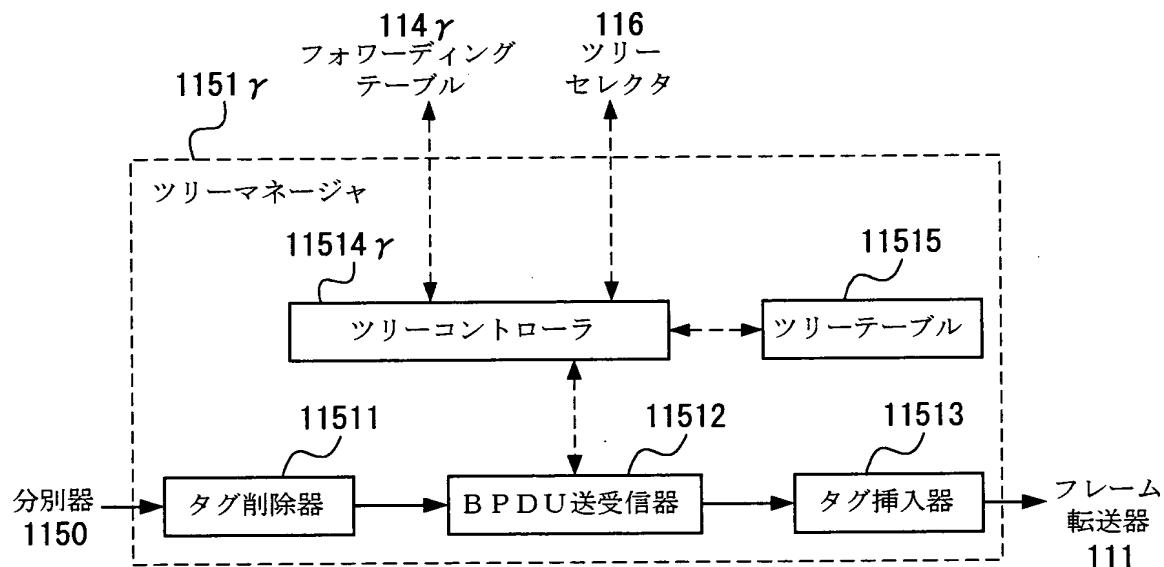


【図4 2】

114γ フォワーディングテーブル  
(ノード11から見たビュー)

タグ1141	出力ポート1142
41	END
42	リンク21側
43	リンク21側
44	リンク23側
45	リンク23側
46	リンク21側

【図4 3】



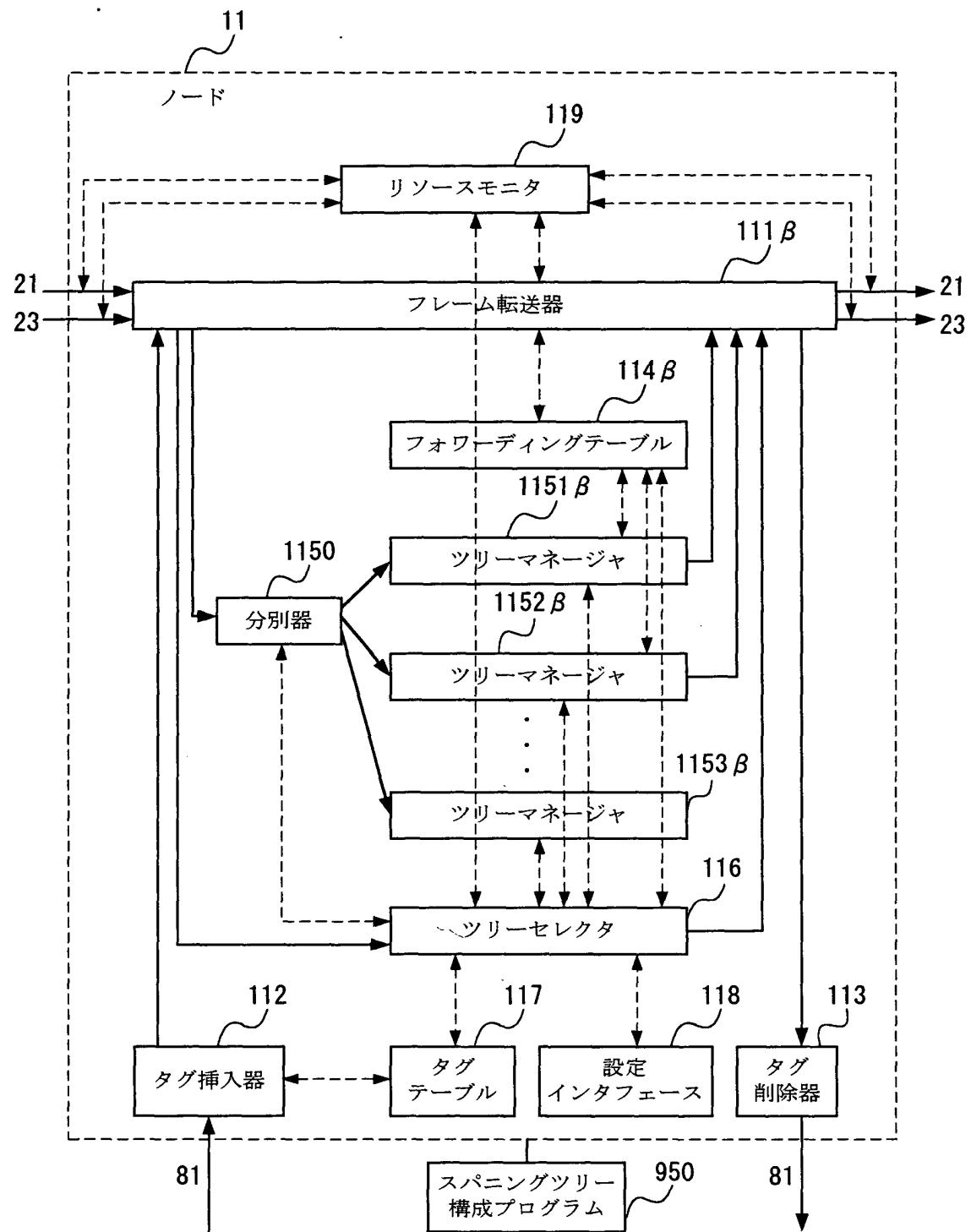
【図44】

スパンニングツリー61(タググループ41)の設定状況



ノード6101	ルートポート6102	タグ1141	出力ポート1142
11	なし	41	END
12	リンク21側	41	リンク21側
13	リンク22側	41	リンク22側
14	リンク23側	41	リンク23側
15	リンク26側	41	リンク26側
16	リンク27側	41	リンク27側

【図45】



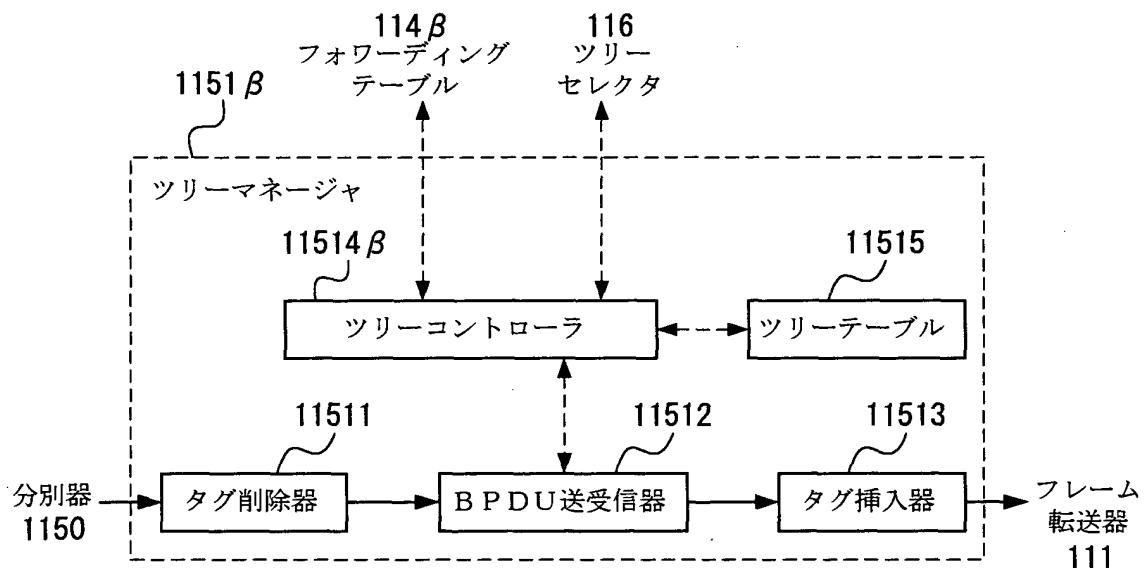
【図4 6】

114 $\beta$  フォワーディングテーブル

(ノード11から見たビュー)

タグ1141	出力ポート1142	ブロードキャスト出力ポート1144
41	タグ削除器側 (END)	リンク21側、リンク23側 (END)
42	リンク21側	リンク23側 (END)
43	リンク21側	— (END)
44	リンク23側	リンク21側 (END)
45	リンク23側	— (END)
46	リンク21側	— (END)

【図4 7】



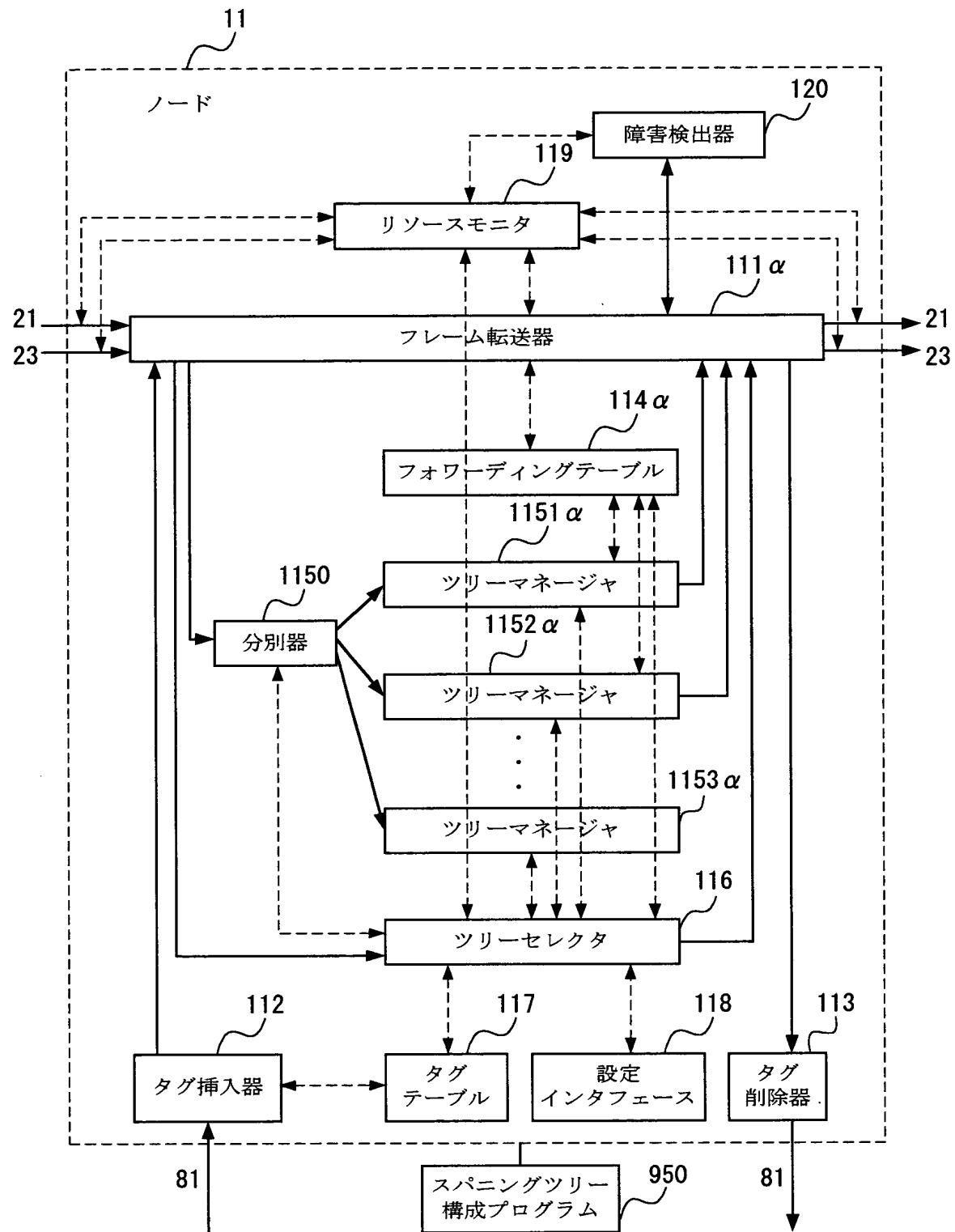
【図48】

スパンニングツリー61(タググループ41)の設定状況



ノード 6101	ルートポート 6102	ディジグネイ テッドポート 6104	タグ 1141	出力ポート 1142	プロードキャスト 出力ポート 1144
11	なし	リンク21側、 リンク23側	41	タグ削除器側	リンク21側、 リンク23側
12	リンク21側	リンク22側	41	リンク21側	リンク22側
13	リンク22側	なし	41	リンク22側	—
14	リンク23側	リンク26側	41	リンク23側	リンク26側
15	リンク26側	リンク27側	41	リンク26側	リンク27側
16	リンク27側	なし	41	リンク27側	—

【図49】

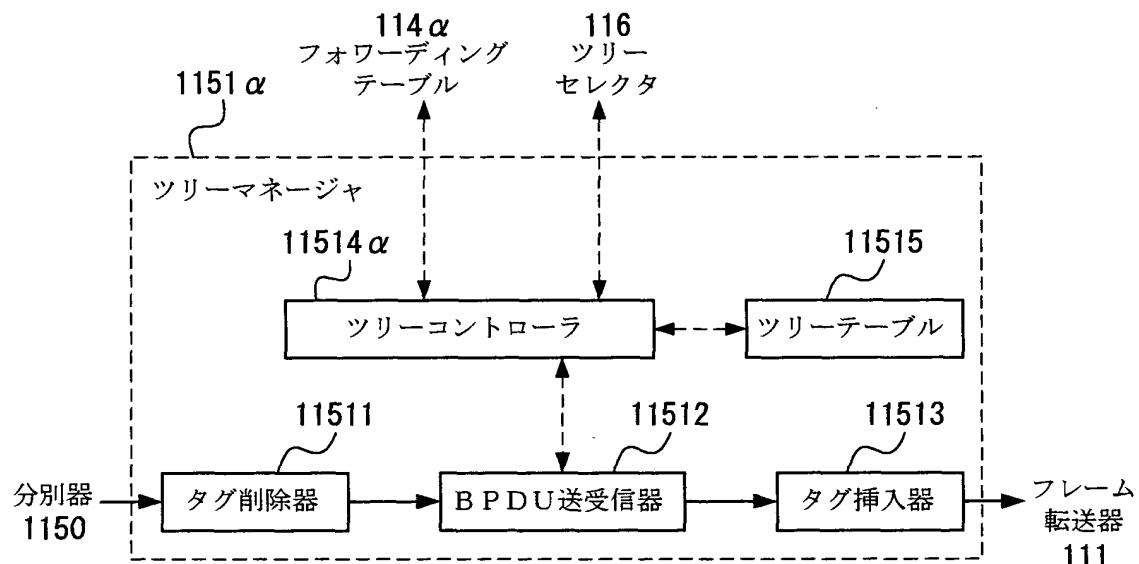


【図50】

114 $\alpha$  フォワーディングテーブル  
(ノード12から見たビュー)

タグ1141	ルートポート1142	予備出力ポート1143
41	リンク21側	リンク24側
42	END	END
43	リンク22側	リンク24側
44	リンク21側	リンク24側
45	リンク24側	リンク21側(又は22)
46	リンク22側	リンク24側

【図51】



【図52】

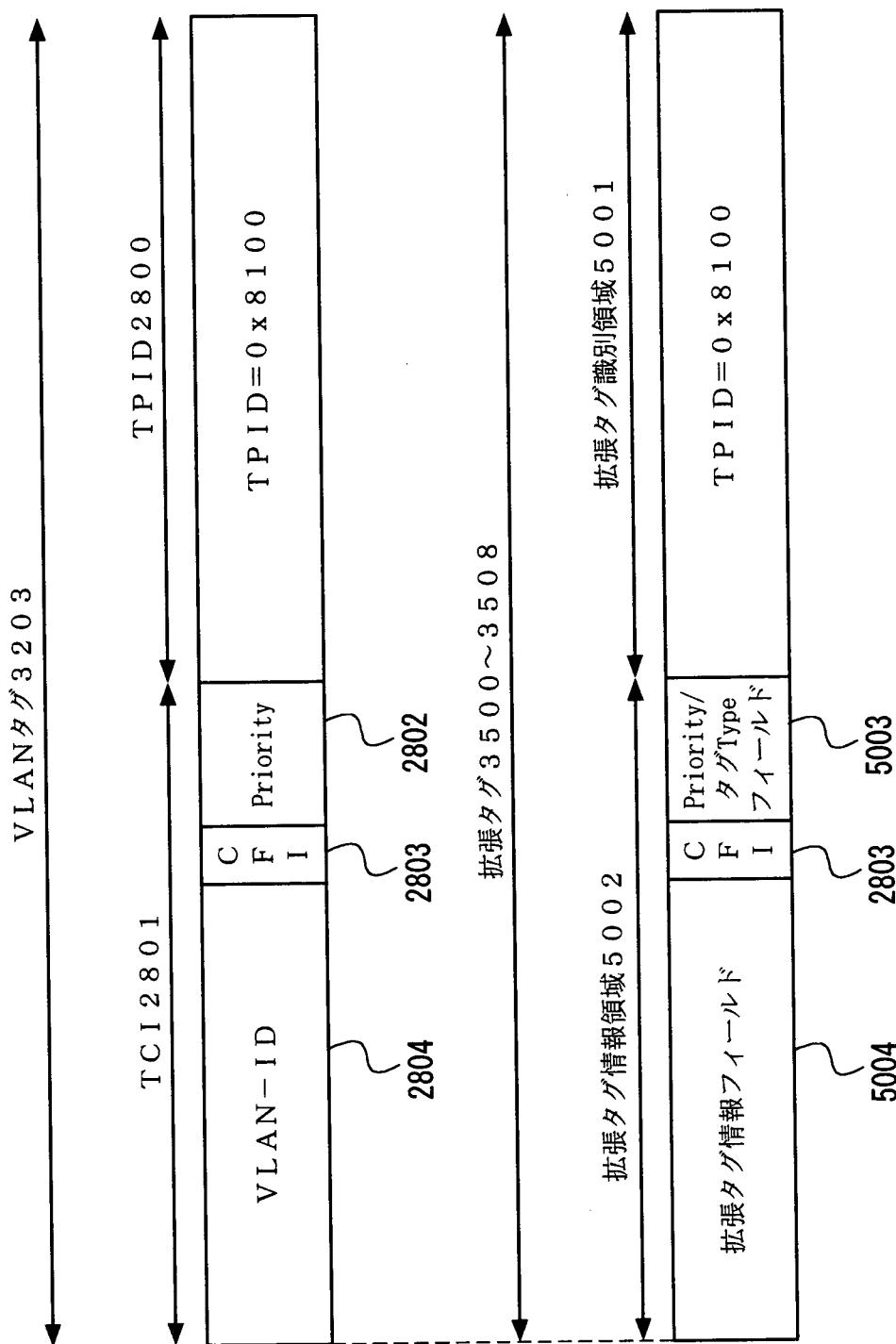
スパンニングツリー61(タググループ41)の設定状況



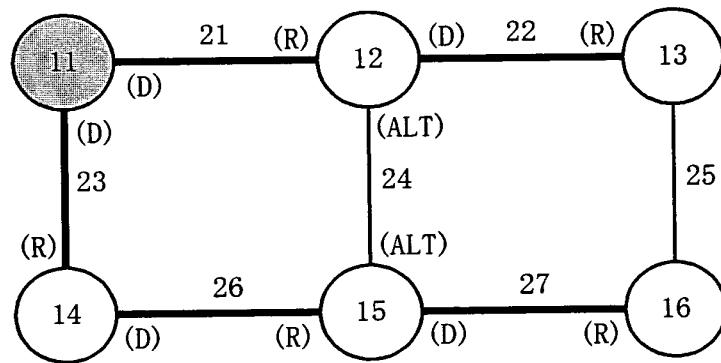
ノード 6101	ルートポート 6102	オルタネート ポート 6103	タグ 1141	出力ポート 1142	予備出力ポート 1143
11	なし	なし	41	タグ削除器側	—
12	リンク21側	リンク24側	41	リンク21側	リンク24側
13	リンク22側	なし	41	リンク22側	—
14	リンク23側	なし	41	リンク23側	—
15	リンク26側	リンク24側	41	リンク26側	リンク24側
16	リンク27側	なし	41	リンク27側	—

^

【図53】



【図54】



【書類名】 要約書

【要約】

【課題】 輻輳発生の確率を下げ、輻輳によるフレームの到着遅れや欠落が発生する頻度を減らすことができるネットワークシステム、スパニングツリー構成方法を提案する。

【解決手段】 複数のノードを接続したネットワーク上でスパニングツリーを構成するノードが、変更前のツリーを運用したまま、別のVLANを用いてコスト変更後のツリーを生成し、新規ツリーの安定後に転送に利用するツリーを切り替える。

【選択図】 図7

認定・付加情報

特許出願の番号	特願2003-041838
受付番号	50300266978
書類名	特許願
担当官	第八担当上席 0097
作成日	平成15年 5月 2日

＜認定情報・付加情報＞

【提出日】 平成15年 2月19日

次頁無

出願人履歴情報

識別番号 [000004237]

1. 変更年月日 1990年 8月29日  
[変更理由] 新規登録  
住 所 東京都港区芝五丁目7番1号  
氏 名 日本電気株式会社